IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 February 2022, accepted 26 April 2022, date of publication 25 May 2022, date of current version 6 July 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3177783

Model-Agnostic Counterfactual Explanations in Credit Scoring

XOLANI DASTILE[1], TURGAY CELIK[2,3,4], AND HANS VANDIERENDONCK[5], (Senior Member, IEEE)

[1]School of Computer Science and Applied Mathematics, University of the Witwatersrand, Johannesburg 2000, South Africa
[2]School of Electrical and Information Engineering, University of the Witwatersrand, Johannesburg 2000, South Africa
[3]Wits Institute of Data Science, University of the Witwatersrand, Johannesburg 2000, South Africa
[4]Faculty of Engineering and Science, University of Agder, 4630 Kristiansand, Norway
[5]School of Electronics, Electrical Engineering and Computer Science, Queen's University Belfast, Belfast BT9 5BN, U.K.

Corresponding author: Turgay Celik (celikturgay@gmail.com)

ABSTRACT The past decade has shown a surge in the use and application of machine learning and deep learning models across various domains. One such domain is credit scoring, where applicants are scored to assess their creditworthiness for loan applications. It is essential to ensure that no biases or discriminations are incurred during the scoring process. Most machine learning and deep learning models are prone to unintended bias and discrimination in the datasets. Therefore, it is imperative to explain each prediction from the models during the scoring process to avoid the element of model bias and discrimination. Our study proposes a novel optimization formulation that generates sparse counterfactual explanations via a custom genetic algorithm to explain the black-box model's predictions. We evaluated the efficacy of the proposed method on publicly available credit scoring datasets by comparing the counterfactual explanations generated by the proposed method with explanations from credit scoring experts. The proposed counterfactual explanation method does not only explain rejected loan applications but also can be used to explain approved loan applications.

INDEX TERMS Credit scoring, machine learning, counterfactual explanation, explainable AI, genetic algorithm.

I. INTRODUCTION

The pervasiveness and ubiquitous nature of machine learning and deep learning models brings about positive change in society with the risk of unintended consequences such as algorithmic bias. The algorithms are not intrinsically biased but inherit the bias from human activities captured in the data during the training process. Thus, the models need to be transparent at all costs. The Basel Accord [1] requires explanations for denied loan applications to ensure that transparency is maintained in automated decisions in the financial sector. Emerging regulations such as the European Union General Data Protection Regulation (GDPR) [2] stipulate that, for automated decisions, a ''right to explanation'' needs to be maintained. Explaining predictions is crucial for high-stake decisions, such as the presence or absence of a disease in the healthcare sector, the rejection or acceptance of a loan application in the finance sector, and the denial or approval of parole in the criminal justice sector. Hence, our study focuses on using a counterfactual explanation technique, which is an instance-based explanation, for explaining predictions from black-box models. Wachter et al. [3] highlighted three key benefits of using the counterfactual explanation, (1) to inform and assist applicants in understanding why a certain decision was made, (2) to give grounds to contest unfair decisions, and (3) to understand what could be changed to attain a desired outcome in the future. A typical example of a counterfactual explanation is a loan application [4], [5]: *"Imagine you filed a credit application at a bank. Unfortunately, the bank rejects your application. Now, you would like to know why. In particular, you would like to know what would have to be different so that your application would have been accepted. A possible explanation might be that you would have been accepted if you would earn 500$ more per month and if you would not have a second credit card."*

Therefore, it is imperative to explain each prediction from the models during the scoring process to avoid the element of model bias and discrimination. We propose a novel optimization formulation that generates sparse counterfactual explanations via a custom genetic algorithm to explain the

The associate editor coordinating the review of this manuscript and approving it for publication was Long Wang.

black-box model's predictions. Our contributions are as follows: 1) a novel formulation of the optimization problem that leads to a single sparse counterfactual explanation using a custom genetic algorithm; 2) a computationally efficient generation of counterfactuals achieved by the normalization of continuous features and selection of predictive features; and 3) validation of automatically generated counterfactuals with the credit scoring experts. Without loss of generality, we used the terms ''counterfactual'' and ''counterfactual explanation'' interchangeably. In addition, the words ''sample'', ''instance'' and ''record'' are used interchangeably.

The remainder of this paper is organized as follows. Section II discusses background and related work on counterfactual explanations. Section III introduces our proposed method for generating counterfactual explanations. The experiment setup is described in Section IV and the results are given in Section V. Finally in Section VI, we summarize the paper and discuss our future work.

II. BACKGROUND AND RELATED WORK

The literature has done a great amount of work on eXplainable Artificial Intelligence (XAI), an emerging artificial intelligence branch. The main purpose of XAI is to make deep learning and machine learning models interpretable. A black-box model can be explained by either using a *post-hoc*, an *ante-hoc* or an *instance-based* explanation. A post-hoc explanation uses another model such as a linear regression or a decision tree to explain the behaviour of a black-box model (e.g. Local Interpretable Model-Agnostic Explanation (LIME) [6]). On the other hand, an ante-hoc explanation is an inherently interpretable model (e.g. Bayesian Rule List [7]), and an instance-based explanation uses an instance to explain the behaviour of a black-box model (e.g. Visual Counterfactual Explanations (ViCE) [8]). Rudin [9] posited that inherently interpretable models (i.e. ante-hoc explanations) should be used for high-stake decisions in lieu of explainable machine learning (referring to post-hoc explanations). The study argued that explainable machine learning generates explanations that are not faithful to what the black-box model computes. Hence, to avoid this shortcoming, this study proposes the use of counterfactuals to explain black-model predictions.

Although counterfactuals seem to produce intuitive explanation systems, some problems remain. Most counterfactual explanation methods generate more than one counterfactual explanation, and this problem is referred to as *Rashomon effect* [5]. The generated explanations might have contradicting ''paths'' on how a certain output was reached. It becomes more difficult and unclear for the user or applicant if there is more than one option of explanations to select from. The other issue with counterfactual explanations is the time it takes to generate the counterfactuals [10]. The time metric can be measured as an average time taken over the generation of a counterfactual for a group of records or the generation of multiple counterfactuals for a single input record [10].

Despite these problems, the research in recent years has shown an increase in the number of studies that focused on explaining machine learning predictions using counterfactual explanations. Grath et al. [11] proposed two weighted approaches to produce counterfactuals for credit scoring data, where one approach derives weights from feature importance, and the other depends on nearest neighbours. Empirical results showed that the weights that are produced from feature importance result in more compact counterfactuals. Furthermore, the study produced positive counterfactuals for accepted loan applications to assist individuals when they make future financial decisions.

In most cases, counterfactuals are generated by using a metaheuristic approach (i.e. an optimization approach that is not problem/task dependent). Guidotti et al. [12] proposed a method that learns a local and interpretable classifier on a synthetic neighbourhood of a record of interest (i.e. an instance that its prediction needs to be explained). The synthetic neighbourhood is generated by a genetic algorithm. The proposed method produces a local explanation that consists of logical rules explaining a decision of the instance of interest and a set of counterfactuals that suggest changes in the instance of interest to produce a desirable outcome. The results showed that the proposed method outperforms previous methods with regards to the quality of the produced explanations and the faithfulness to the black-box model. Sharma et al. [13] proposed a unified and model agnostic approach to address non-transparency (among other issues) of black-box models by using counterfactuals that are generated via a genetic algorithm. The proposed approach achieved robustness, transparency, interpretability, and fairness of black-box models. Further, the study intends to improve the speed of genetic algorithms in their future work. Sharma et al. [13] is the closest study to our approach.

Once the counterfactuals are generated, the next thing to look at is the feasibility of the generated counterfactuals. A change in a few features makes counterfactuals to be feasible. Van Looveren et al. [14] proposed a framework for generating sparse and in-distribution counterfactuals. A sparse counterfactual refers to a counterfactual that requires minimum feature changes to belong to the desired class.

Poyiadzi et al. [15] argued that the current methods that generate counterfactuals for explanation are not considering the feasibility of the generated counterfactuals in the real world. This is attributable to counterfactuals that do not represent the underlying data distribution. The study proposed a method that generates feasible and actionable counterfactual explanations based on the shortest path distance determined by density-weighted metrics. The proposed approach generates counterfactuals that are logical and consistent with the underlying data distribution, making counterfactuals feasible and actionable. Mothilal et al. [16] posited that counterfactuals should be feasible and diversified. The study proposed a framework for generating and assessing the diversity of counterfactuals based on a determinant of a kernel matrix.

The proposed framework generates diverse counterfactuals as opposed to previous methods.

Efficiency and speed are key factors when generating counterfactuals. It is better to use fewer computer resources to speed up the computation time. Resource-intensive methods tend to result in high computation time. Artelt and Hammer [17] investigated how to efficiently compute counterfactuals for prototype-based classifiers. The study discovered that in most cases, either a set of linear or convex quadratic programs that generate counterfactuals can be solved efficiently. Van Looveren and Klaise [18] proposed the use of class prototypes to speed up the generation of counterfactuals. The class prototypes are either attained by employing an encoder or class-specific *k-d* trees. Efficiency is synonymous with the generation time of counterfactuals.

Not only speed and efficiency are essential when generating counterfactuals, but the safety of the black-box models. Counterfactuals guarantee the safety of the black-box models. Sokol and Flach [19] showed that when making AI systems explainable, there is a chance of compromising the safety and security of the system and the possibility of data leakage. This poses a challenge to Explainable AI systems, and the study suggests that the security of AI systems will not be compromised when counterfactuals are used in lieu of other explainable AI techniques.

In general, the credit scoring literature is not extensive when it comes to counterfactual explanations. Hence, this study aims to expand the use of counterfactual explanations in credit scoring to address the issue of multiple counterfactuals that are generated to explain a single instance. Further, this study aims to suggest alternative ways to deal with the efficiency and sparseness of counterfactuals.

III. METHODOLOGY

This section outlines the methodology that is undertaken in this study. The key design decisions were to collect the datasets, select predictive features, calculate correlations between the target variable and the predictors, normalize each continuous feature of the datasets, formulate an optimization problem that will help to generate the counterfactuals, measure the time it takes to generate the counterfactuals, and to compare the generated counterfactuals with experts' opinions. Each of the above steps helped us to obtain counterfactuals that are sparse, generated fast, and robustly tested.

Figure 1 shows the flowchart of our proposed methodology. Firstly, we focus on feature selection, where we select predictive features using a random forest model. Secondly, we calculate Spearman's correlation coefficient between the target variable and the features. The aim of calculating the correlations is to create sparse counterfactuals. This means that we will only focus on features that are better correlated with the target variable when generating the counterfactuals. The feature selection together with sparsity ties back to the feasibility of the counterfactuals, which requires a minimal number of features to be changed. Thereafter, the dataset is split into categorical and continuous features, and the continuous features are normalized. The purpose of normalizing continuous features is to ensure that the features have the same scale. This allows counterfactuals to be generated much quicker because there is no huge varying degree of values within each feature. We proceed by merging the categorical and continuous features. The dataset is then split using the K-fold cross validation and a classifier is trained and its performance is assessed. The purpose of the K-fold cross validation is to ensure that model robustness is maintained during training. The final step is to generate a counterfactual that will explain the predicted outcome for a data point that we are interested in (i.e. a defaulted loan).

A. DATA PREPROCESSING

We selected predictive features via a random forest. The random forest is an ensemble of decision trees. At each node of the decision tree, a split is determined by the measure of impurity of each feature, either by using the *entropy* or the *gini index*. The importance or predictive power of each feature is derived from the impurity calculation, which is shown by either the entropy

$$I_E = -\sum_{j=0}^{1} p_j \log_2 p_j \tag{1}$$

or the gini index

$$I_G = 1 - \sum_{j=0}^{1} p_j \tag{2}$$

where p_j represents proportion of samples in each class. The more "pure" a feature is, the more important it is. The importance of each feature is obtained from how "pure" a feature is. The more the feature reduces impurity, the more important the feature is.The final importance of the feature is the average impurity decrease across all decision trees. Thus, all selected features will have a maximum average impurity decrease across all decision trees.

B. COUNTERFACTUAL SPARSITY

To ensure that the counterfactuals that are generated are sparse, we used Spearman's rank correlation coefficient. Spearman's rank correlation coefficient is based on the rankings of a feature as opposed to using raw feature values. The benefit of using rankings is that both continuous and categorical features can be used to calculate the correlation. Hence, sparsity is determined by the correlation between the target variable (which is categorical in nature) and the predictors (which are either categorical or continuous). Spearman's rank correlation coefficient is given as

$$\rho = 1 - \frac{6 \times \sum z_i^2}{n(n^2-1)} \tag{3}$$

where z_i is the difference between two ranks of each record and n is the number of records. The aim of calculating the correlation is to focus only on features that are correlated with the target variable when the counterfactuals are generated.

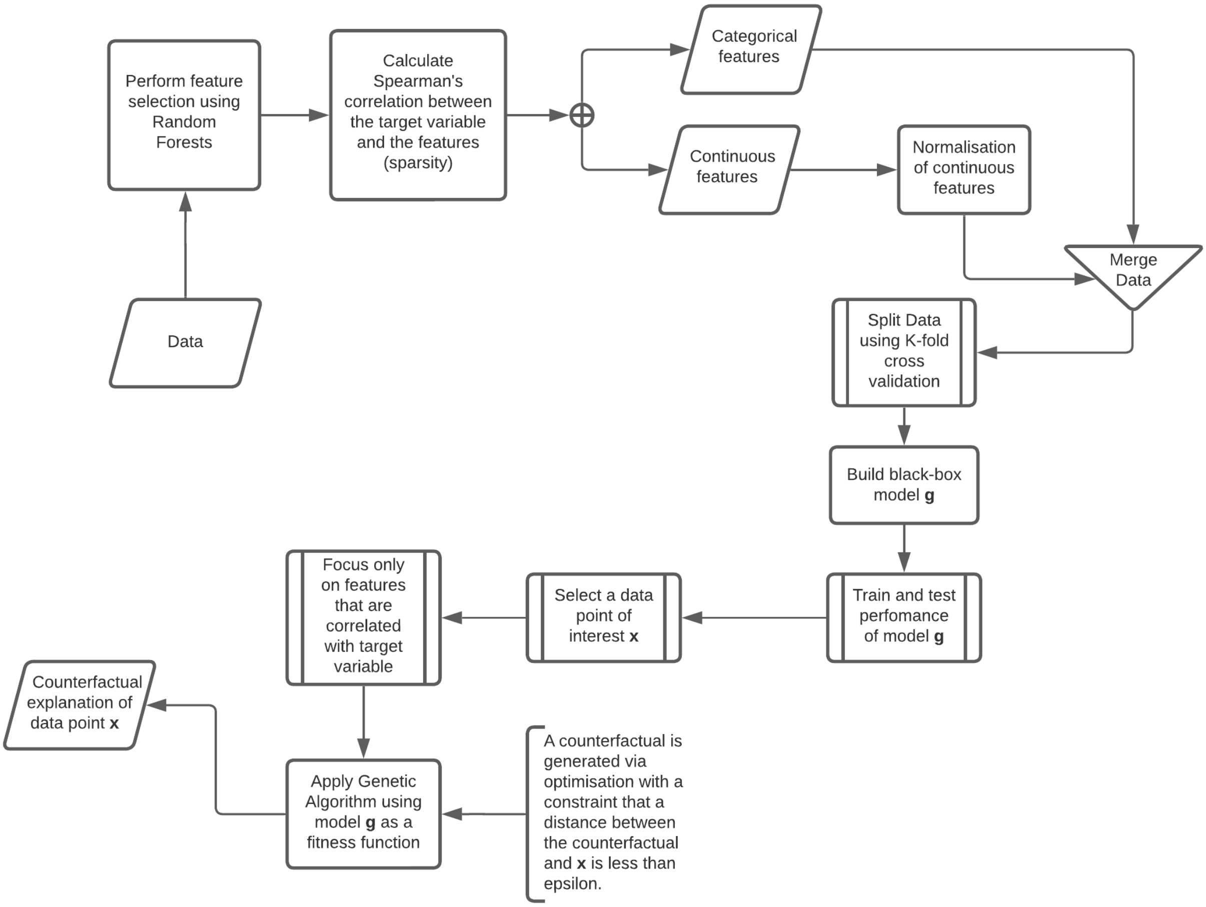
Perform feature selection using Random Forests
Calculate Spearman's correlation between the target variable and the features (sparsity)
Categorical features
Continuous features
Normalisation of continuous features
Merge Data
Data
Split Data using K-fold cross validation
Build black-box model **g**
Train and test perfomance of model **g**
Select a data point of interest **x**
Focus only on features that are correlated with target variable
Counterfactual explanation of data point **x**
Apply Genetic Algorithm using model **g** as a fitness function
A counterfactual is generated via optimisation with a constraint that a distance between the counterfactual and **x** is less than epsilon.

FIGURE 1. The flowchart of the proposed methodology.

The statistical significance of a Spearman's rank correlation is tested by using a hypothesis testing. The test will measure the association between the target variable and the predictors. The null hypothesis and the alternative hypothesis are stated as

- H_0 = There is no monotonic association between variables.
- H_a = There is a monotonic association between variables.

The level of significance for the hypothesis test is $\alpha = 5\%$. The null hypothesis H_0 is rejected when the p-value $< \alpha$. If the null hypothesis is rejected, then the conclusion will be that there is statistical evidence that there is association between variables.

C. NORMALIZATION OF CONTINUOUS FEATURES

Let $\mathbf{x}_k \in \mathbb{R}^d$ denote a feature vector for record k, where d represents the number of features in a dataset. The labeled dataset is represented by $\mathbf{X} = \{(\mathbf{x}_k, y_k)\}_{k=1}^{n}$, where $y_k \in \{0, 1\}$ represents the target/response flag and $\mathbf{x}_k = \{f^i\}_{i=1}^{d}$ is the feature vector, and n denotes the number of records in the dataset. Let $\mathbf{f}^i$ represent each feature in $\mathbf{X}$, $\forall i \in \{1, 2, \cdots, d\}$. Continuous feature values were converted by using a normalization technique

$$f^i_{norm} = \frac{f^i - \min(\mathbf{f}^i)}{\max(\mathbf{f}^i) - \min(\mathbf{f}^i)}, \tag{4}$$

such that

$$f^i_{norm} \in [0, 1], \tag{5}$$

where $\min(\mathbf{f}^i)$ and $\max(\mathbf{f}^i)$ are the minimum and the maximum values for feature i, respectively.

D. GENETIC ALGORITHM

The genetic algorithm is a heuristic search approach that is motivated by natural evolution [20]. The genetic algorithm searches for optimal values that can either minimize or maximize a certain function. Hence, genetic algorithms are used for optimization problems. The genetic algorithm involves a selection of individuals (i.e. parents) (based on some defined fitness function) from an initial population for reproduction. Individuals that are unfit for reproduction are omitted. The parents produce an offspring that inherits their (i.e. parents) characteristics. The offspring then gets included in the next generation of the population. The process repeats itself until it converges into a solution. The genetic algorithm process involves five phases i.e. the *initial population*, the *fitness function*, the *selection*, the *cross-over*, and the *mutation*. In the context of credit scoring, a population individual is a feature vector and this feature vector is regarded as a counterfactual. To select the best counterfactuals, a fitness function is used. The fitness function in this study is the black-box model that is required to be explained by the counterfactual. The fitness function output is probabilistic and has values in the range [0, 1]. The quality of each counterfactual is determined by the output of the fitness function and the distance between the counterfactual and the feature vector of interest (i.e. a feature vector with a default response flag) should be less than some *epsilon* value. Selected counterfactuals go to a mating pool and these counterfactuals are known as parents. Every two parents will produce two offspring (i.e. two counterfactuals). The mating process is a cross-over phase. The cross-over is based on a cross-over rate which is defined as the probability of two parents crossing over at a single point [20]. Mating high-quality parents will generate better-quality offspring with similar traits as the parents. This removes bad population individuals from generating more bad individuals. Note that, the offspring will have similar drawbacks as their parents. To overcome the drawbacks, some changes will need to be made to the offspring to generate new offspring. The changes are known as the mutation phase. The mutation is responsible for randomly changing values in the counterfactual based on a mutation rate which is defined as the probability of determining the number of counterfactuals that should be mutated in a single population [21]. The main purpose of mutation is to preserve diversity in a population, and this prevents early convergence to a solution.

E. FORMULATION OF THE OPTIMIZATION PROBLEM

This section defines our formulation of the optimization problem which is solved via a custom genetic algorithm. Let $\mathbf{c}^* \in \mathbb{R}^d$ denote a counterfactual for record $\mathbf{x} \in \mathbb{R}^d$ and g denote a black-box model. Let g have a probabilistic cartesian product output. Then g is given as

$$g : \mathbb{R}^d \to A \times B, \tag{6}$$

where

$$A \times B = \{(a, b) | a \in [0, 1], b \in [0, 1], a + b = 1\} \tag{7}$$

and a represents a probability of belonging to class 0 and b is a probability of belonging to class 1. The aim is to find $\mathbf{c}^* \in \mathbb{R}^d$ such that

$$\mathbf{c}^* = \underset{\mathbf{c}\in\mathbb{R}^d}{\arg\min}\ g_{b\in[0,1]}(\mathbf{c}) \tag{8}$$

subject to

$$\text{dist}(\mathbf{c}^*, \mathbf{x}) < \epsilon \tag{9}$$

and

$$c^{*,i} \in [\min(\mathbf{f}^i), \max(\mathbf{f}^i)], \quad \forall i \in \{1, 2, \cdots, d\}. \tag{10}$$

The main idea behind the above optimization problem is to find optimal $\mathbf{c}^* \in \mathbf{C}$ (i.e. $\mathbf{C}$ is the set of counterfactuals) that result in a non-default class, ensuring that the generated counterfactual $\mathbf{c}^*$ is as close as possible to the instance of interest $\mathbf{x}$ and that $\mathbf{c}^*$ comes from the same data distribution as $\mathbf{x}$. The output of each class (i.e. default or non-default) is

$$\text{class output} = \begin{cases} 0, & \text{if } (a, b) \in [0.5, 1] \times [0, 0.5) \\ 1, & \text{if } (a, b) \in [0, 0.5) \times [0.5, 1]. \end{cases} \tag{11}$$

In Eq. (8), we ensure that $b \in [0, 1]$ is minimized so that $a \in [0.5, 1]$ makes the counterfactual to belong to a non-default class. Please note that class output $= 0$ and class output $= 1$, means that $\mathbf{x}$ belongs to a non-default class and default class, respectively. According to Wachter et al. [3], closeness of $\mathbf{c}^*$ to $\mathbf{x}$ should be measured by the L_1 norm distance divided by the mean absolute deviation (MAD). Wachter et al. [3] showed that the L_1 norm distance divided by MAD is better than the L_1 or L_2 norm distances. The first part of Equation (12) is for continuous features and the latter part is for categorical features. Since we are treating categorical values as discrete, hence we used the mean deviation which is suitable for discrete values. The choice of distance in our study is

$$\text{dist}(\mathbf{c}^*, \mathbf{x}) = \sum_{i=1}^{d^*} \frac{|c^{*,i} - f^i|}{MAD^i} + \sum_{i=d^*+1}^{d} \frac{|c^{*,i} - f^i|}{MD^i}, \tag{12}$$

where MAD^i and MD^i denote a mean absolute deviation and a mean deviation for feature i, respectively, and d^* denotes the number of continuous features. The mean absolute deviation for feature i is

$$MAD^i = \frac{1}{n}\sum_{j=1}^{n} |f_j^i - \bar{\mathbf{f}}^{(i)}|, \tag{13}$$

where n denotes the number of records and $\bar{\mathbf{f}}^{(i)}$ denotes the mean or average of feature i. The mean deviation for feature i is

$$MD^i = \frac{1}{m}\sum_{l=1}^{m} o_l |f_l^i - \tilde{\mathbf{f}}^{(i)}|, \tag{14}$$

where m is the number of categories, o_l is the frequency of each category and $\tilde{\mathbf{f}}^{(i)}$ is the median of feature i.

The threshold for the chosen distance is defined by ϵ which is provided by the user. The fitness function in our optimization problem is $g(\mathbf{c}^*)$. To explain predictions of applicants that are approved for loans (i.e. class output $= 0$), the aim is to find optimal values of $\mathbf{c}^*$ such that

$$\mathbf{c}^* = \underset{\mathbf{c}\in\mathbb{R}^d}{\arg\min}\ g_{a\in[0,1]}(\mathbf{c}) \tag{15}$$

subject to the same constraints of Equation (8). In Eq. (15), we ensure that $a \in [0, 1]$ is minimized so that $b \in [0.5, 1]$ makes the counterfactual to belong to a default class.

The major differences between our study and that of Sharma et al. [13] are 1) the formulation of the optimization problem, 2) the distance calculation for categorical features and 3) the generation of sparse counterfactuals. Sharma et al. [13] used the following fitness function,

$$\text{fitness function} = \frac{1}{\text{dist}(\mathbf{c}^*, \mathbf{x})} \tag{16}$$

where

$$\text{dist}(\mathbf{c}^*, \mathbf{x}) = \frac{n_{con}}{n}\sum_{i=1}^{n_{con}}\frac{|c^{*,i} - f^i|}{MAD^i} + \frac{n_{cat}}{n}\text{SimpleMat}({\mathbf{c}^*}_{cat}, \mathbf{x}_{cat}) \tag{17}$$

and n_{con} and n_{cat} denote the number of continuous and categorical features, respectively, and ${\mathbf{c}^*}_{cat}$ and $\mathbf{x}_{cat}$ are categorical attributes for the counterfactual $\mathbf{c}^*$ and record $\mathbf{x}$, respectively. The simple matching coefficient is used in Sharma et al. [13] to deal with categorical features and is given as

$$\text{SimpleMat} = \frac{\text{number of matching attributes}}{\text{total number of attributes}}, \tag{18}$$

and has values between 0 and 1.

IV. EXPERIMENTS

A. DATA

Publicly available credit scoring datasets were used in this study, i.e. German [22] and Home Loan Equity (HMEQ) [23]. The German dataset can be accessed on the *UCI* repository, and the HMEQ dataset can be accessed on *Kaggle*. The German credit dataset has 20 features, where 7 of the features are numerical and the other 13 are categorical. The German credit dataset has features such as `status of existing checking account`, `duration in month`, `credit history` and `purpose` to mention a few. The HMEQ dataset has 13 features, where 11 of the features are numerical and the other 2 are categorical. The features are the `amount of loan request`, `amount due on existing mortgage`, `value of the current property`, `years at present job`, `number of credit lines` to mention a few. The `target variable` for each of the above credit scoring datasets is binary, i.e. applicants are classified either as "default" (i.e. bad applicants denoted by a 1) or "non-default" (i.e. good applicants denoted by a 0).

B. DATA SPLIT, MODEL TRAINING AND MODEL PERFORMANCE

For robustness of the classifier g, a K-fold cross validation [24] was used. The K-fold cross validation splits the data into K equal folds $\mathcal{D}_1, \mathcal{D}_2, \cdots, \mathcal{D}_K$, where

$$\bigcup_{k=1}^{K}\mathcal{D}_k = \mathbf{X}. \tag{19}$$

Each fold $\mathcal{D}_k$ is used as a test set and the remaining $\mathcal{D} = \{\mathcal{D}_j\}_{j=1, j\neq k}^{(K-1)}$ folds are used for model training. The overall performance metric for the model is

$$g_p = \frac{1}{K}\sum_{k=1}^{K} g_p^k \tag{20}$$

and g_p represents the average model performance metric, e.g. accuracy or Area Under the Curve (AUC), and g_p^k is a model performance metric in each of the test folds $\mathcal{D}_k$. Note that the remaining folds $\mathcal{D} = \{\mathcal{D}_j\}_{j=1, j\neq k}^{(K-1)}$ were used in turn for training, we did not build K distinct models.

C. EXPERIMENT SETUP

For the genetic algorithm, we ran four experiments on each dataset to select optimal parameters based on the execution times of the optimization problem, the resulting class output and the distance between the record that needs to be explained and the generated counterfactual. Table 1 shows different parameter values that were used for the genetic algorithm. We then measured the times in seconds for the execution of the optimization problem, the resulting class outputs, and the distances were also assessed and are shown in Table 1. For German dataset, we chose $\epsilon = 4$, mutation rate $= 0.01$, cross-over rate $= 0.40$ and population size $= 30$, since they are giving a least amount of execution time, the desired class output (i.e. the non-default output), and the minimal distance. For HMEQ dataset, we chose $\epsilon = 7$, mutation rate $= 0.04$, cross-over rate $= 0.70$ and population size $= 100$, since they are giving the desired class output (i.e. the non-default output) and the least distance. The models that were used in our study were random forest and artificial neural networks (this was to ensure that our approach is model-agnostic). The choice of these models was based on our previous literature review study [25]. The parameters for the random forest model were set as follows, the maximum depth for each decision tree was set to 5, and the number of decision trees was set to 100. For the artificial neural networks, we used 3 hidden layers, the first hidden layer had 50 neurons, and the second hidden layer had 30 neurons and the third hidden layer had 5 neurons, with the output layer having 2 neurons. The parameters for both neural networks and random forest were obtained by using a grid-search approach. To train the models, we used K-fold cross validation, where we set K = 5.

D. EXPERIMENTS FOR CREDIT SCORING EXPERTS

Initially, we contacted about six credit scoring experts and there were only three experts who showed interest in taking part in the experiments of this study. It is challenging to find credit scoring experts since they have busy schedules and are mostly not available. The use of several experts in this study was to ensure that we get different views from industry experts. This will then ensure that the counterfactuals are robustly assessed. All the experts in this study have more than 7 years of working experience in the financial sector, working as quantitative analysts in credit scoring departments. They all have a background in developing credit scorecards from scratch. The experts were given a data dictionary with a

TABLE 1. Selection of parameters for genetic algorithm based on execution times (measured in seconds), the resulting objective function and the distance between the original feature vector and the counterfactual vector on German and HMEQ credit datasets. Legend: time (execution time for optimization problem), func (predicted class output), dist (distance between record that needs to be explained and the counterfactual). For predicted class output, 0 denotes non-default class and 1 denotes default class.

Experiments	German	HMEQ
Experiment 1	$\epsilon = 4$	$\epsilon = 4$
	mutation rate = 0.01	mutation rate = 0.01
	cross-over = 0.40	cross-over = 0.40
	population = 30	population = 30
	time = 386, func = 0, dist=3.75	**time = 60, func = 0, dist=24**
Experiment 2	$\epsilon = 5$	$\epsilon = 5$
	mutation rate = 0.02	mutation rate = 0.02
	cross-over = 0.50	cross-over = 0.50
	population = 40	population = 40
	time = 531, func = 0, dist=3.75	**time = 87, func = 0, dist=15**
Experiment 3	$\epsilon = 6$	$\epsilon = 6$
	mutation rate = 0.03	mutation rate = 0.03
	cross-over = 0.60	cross-over = 0.60
	population = 60	population = 60
	time = 811, func = 0, dist=5	**time = 123, func = 0, dist=195**
Experiment 4	$\epsilon = 7$	$\epsilon = 7$
	mutation rate = 0.04	mutation rate = 0.04
	cross-over = 0.70	cross-over = 0.70
	population = 100	population = 100
	time = 1378, func = 0, dist=3.75	**time = 192, func = 0, dist=6**

detailed description of each feature. The experts were asked to identify features that they deem fit in influencing the prediction of the record of interest (i.e. the record that belongs to a default class) based on their domain expertise. This was to allow credit experts to have a subjective opinion on the features that they thought were key contributors in predicting the class of the record of interest. The credit risk experts then created their counterfactuals and they assessed the prediction of their counterfactuals using an online user interface, to see if the counterfactuals will belong to the desired class (i.e. a non-default class). The experts were also asked to normalize all continuous features before they use the online platform. The online user interface can be found on this link (https://xolani-explanation-research.herokuapp.com/). We created this user interface from scratch and the scoring models that were used in the user interface for predictions are the ones that we are explaining using the counterfactuals in this study.

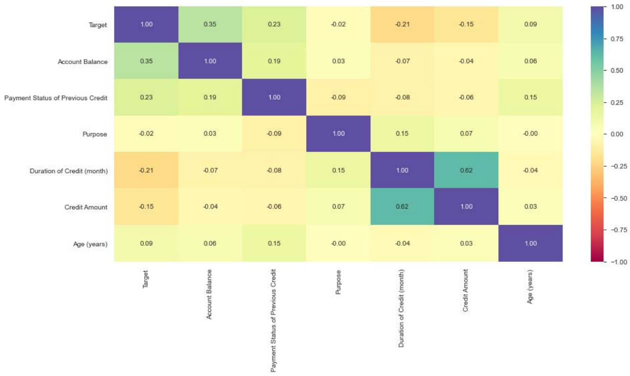

FIGURE 2. Spearman's rank correlation coefficient for the German credit dataset. The selected features are `Account Balance`, `Payment Status of Previous Credit` and `Duration of Credit (month)` based on the correlation coefficients with the `Target` variable. The p-values of the correlation coefficients for all selected features are ≈ 0.

V. RESULTS

This section starts by looking at the features that are correlated with the target variable and thereafter the model performances of the random forest classifier on German credit dataset and the artificial neural networks classifier on HMEQ credit dataset. Lastly, explanations from our approach, Sharma et al. [13] approach and from the experts are examined.

A. COUNTERFACTUAL SPARSITY

We assessed the correlations between the target variable and the predictors. The aim is to focus only on predictors that are better correlated with the target variable when we are generating our counterfactuals. Figure 2 and Figure 3 show correlation matrices for the German and HMEQ credit datasets, respectively. In Figure 2, the predictors that are better correlated with the target variable are `Account Balance` and `Payment Status of Previous Credit`. Please note that by *"better correlation"* we mean that compared to the rest of the predictors, the selected predictors have a *"higher"* correlation with the target variable. In Figure 3, the predictors that are better correlated with the target variable are `DEBTINC`, `DEROG` and `DELINQ`. Thereafter, we tested the statistical significance of the correlations. All the selected features in both datasets that are correlated with the target variable show that the associations are statistically significant since their p-values $\ll 5\%$.

B. MODEL PERFORMANCE

Table 2 shows the classifier performance results that were reported in literature, and we compared those results to the performances of the models that were used in our study.

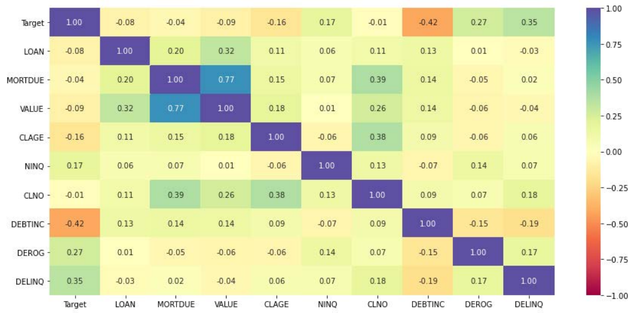

FIGURE 3. Spearman's rank correlation coefficient for the HMEQ credit dataset. The selected features are `DEBTINC`, `DEROG` and `DELINQ` based on the correlation coefficients with the `Target` variable. The p-values of the correlation coefficients for all selected features are ≈ 0.

TABLE 2. Credit scoring model performances that are reported in the literature.

Source	Year	Model	German		HMEQ	
			Accuracy	AUC	Accuracy	AUC
[26]	2000	Logistic Regression	0.76	-	-	-
[27]	2016	Neural Network	0.75	-	-	-
[28]	2017	XGBoost	0.77	-	-	-
[29]	2018	Neural Network	0.78	0.80	-	-
Our Study	2022	Random Forest	0.74	0.76	-	-
Our Study	2022	Neural Network	-	-	0.78	0.79

A detailed comparison of model performances in credit scoring can be found in our previous study [25]. In Table 2, the results do not significantly differ from each other, and this proves the efficacy of our model choice for our study. The main purpose of our study is not to compare model performances but to explain how a model makes its prediction and what actions are required to effect a desired outcome for the loan applicant.

C. EXPLANATIONS

1) PREDICTION EXPLANATIONS FROM OUR APPROACH AND FROM SHARMA et al. [13] APPROACH

Please note that in this study, we also implemented from scratch, the approach that was used in [13]. The explanations are given in Figure 4 and Figure 5, for German and HMEQ credit datasets, respectively. The feature names are on the y-axis, the x-axis represent the change between the original feature value and the counterfactual feature value. Please note that all continuous values in the figures were normalized, however when providing explanations using text, the normalized values were denormalized to make sense out of the explanations. Using our approach on German credit data, the applicant would qualify for the loan if the `Payment Status of Previous Credit` *decreases* by 3 and `Account Balance` *decreases* by 1. Using the approach by Sharma et al. [13] on German credit data, the applicant would qualify for the loan if the `Account Balance` *decreases* by 1, `Credit Amount` *increases* by 15630, `Duration of Credit (month)` *increases* by 43, `Purpose` *increases* by 5 and `Age (years)` *decreases* by 6. Using our approach on HMEQ credit data, the applicant would qualify for the loan if the `DEBTINC` *decreases* by 32, `DELINQ` *increases* by 1 and `DEROG` *increases* by 3. Using the approach by Sharma et al. [13] on HMEQ credit data, the applicant would qualify for the loan if the `LOAN` *increases* by 39072, `MORTDUE` *increases* by 325939, `VALUE` *increases* by 84791, `CLAGE` *increases* by 1051, `NINQ` *increases* by 4, `CLNO` *increases* by 9, `DEBTINC` *decreases* by 24 and `DEROG` *increases* by 10.

TABLE 3. Comparison of our approach with the approach from Sharma et al. [13] using different metrics. Legend: time (execution time for optimization problem measured in seconds), func (predicted class output), dist (distance between the record that needs to be explained and the counterfactual). For predicted class output, 0 denotes non-default class and 1 denotes default class.

	Performance metrics	
	German	HMEQ
Sharma et al. [13]	time=879, func=0, dist= 5	time=722, func=0, dist=16
Our approach	time=386, func=0, dist=3.75	time=192, func=0, dist=6

The difference between our approach and that of Sharma et al. [13] is around the time it takes to generate a counterfactual and the distance between the record of interest and the generated counterfactual. The generated counterfactual must not be far off from the data point of interest, this ensures that the counterfactual is feasible. Our approach uses sparse counterfactuals that result in small distances between the counterfactual and the record of interest. This is illustrated in Table 3.

2) EXPLANATIONS FROM CREDIT SCORING EXPERTS

Credit scoring experts generated their own counterfactual explanations based on their domain knowledge. Please refer to Figure 4 and Figure 5 to visually see the explanations that are given in the text below. Expert number 1 on German credit dataset, stated that the applicant would qualify for the loan if the `Payment Status of Previous Credit` *decreases* by 4, `Duration of Credit (month)` *increases* by 10, `Credit Amount` *increases* by 5702, `Account Balance` *increases* by 1 and `Age (years)` *increases* by 7. Expert number 2 on German credit dataset, suggested that the applicant would qualify for the loan if the `Duration of Credit (month)` *increases* by 57, `Credit Amount` *increases* by 10359 and `Purpose` *decreases* by 1. Expert number 3 on German credit dataset, stated that the applicant would qualify for the loan if the `Duration of Credit (month)` *increases* by 38, `Age (years)` *decreases* by 10 and `Payment Status of Previous Credit` *decreases* by 3. Expert number 1 on HMEQ credit dataset, stated that the applicant would qualify for the loan if the `LOAN` *increases* by 11544,`DEROG` *increases* by 2, `VALUE` *increases* by 373079, `CLAGE` *increases* by 456, `NINQ` *increases* by 8, `CLNO` *increases* by 42 and `DEBTINC` *decreases* by 14. Expert number 2 on HMEQ credit dataset, stated that the applicant would qualify for the loan if the the `CLAGE` *increases* by 362. Expert

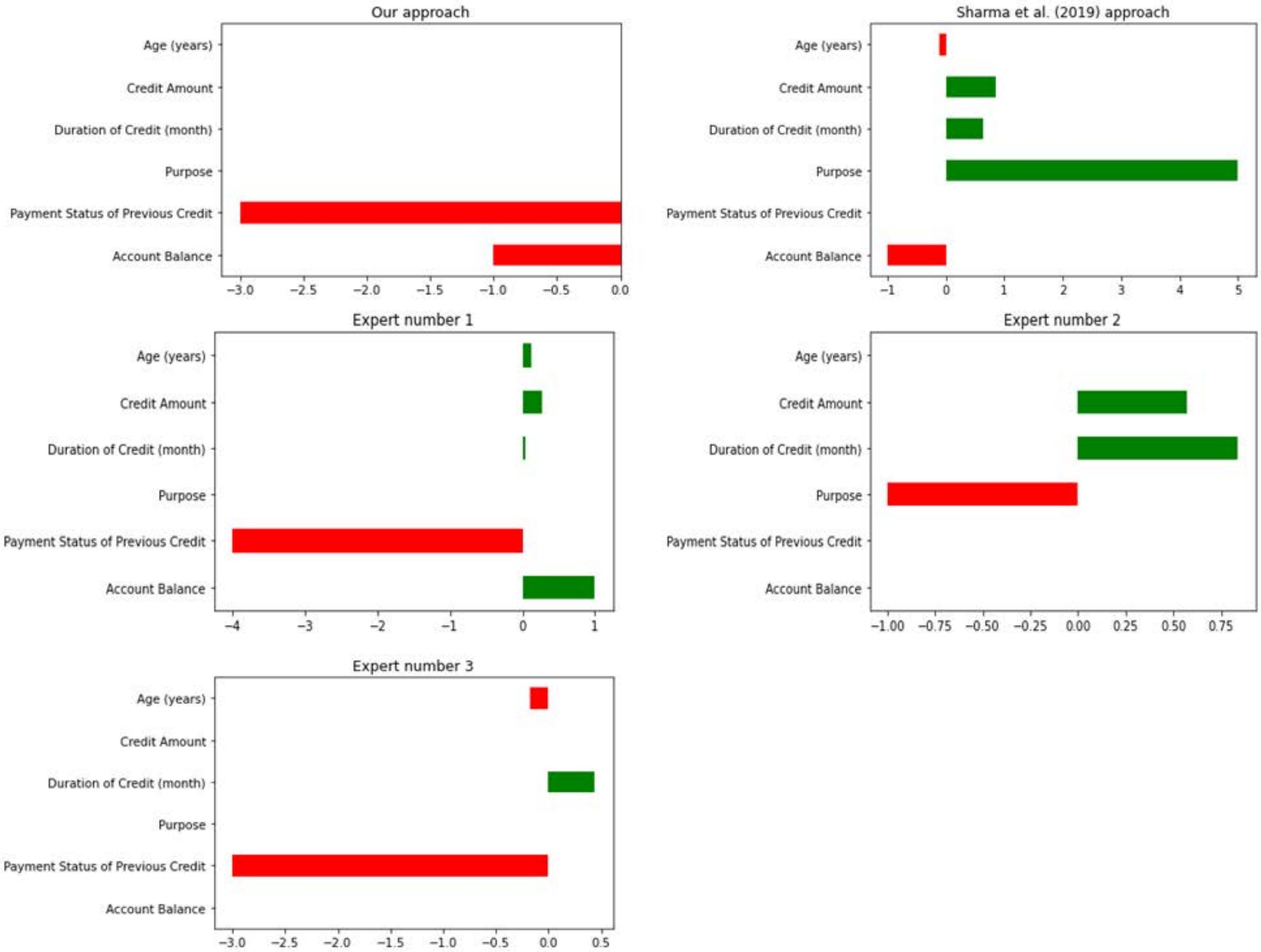

FIGURE 4. **Explanation on German credit dataset record using our approach, Sharma et al. [13] and experts. The red bars represent a decrease in a feature value and the green bars represent an increase in a feature value. The x-axis on each figure represent the change between the original feature value and the counterfactual feature value.**

number 3 on HMEQ credit dataset, stated that the applicant would qualify for the loan if the the `CLAGE` *increases* by 304.

For counterfactuals that were generated by the experts, in some cases the features that needed to be changed overlapped with the features that were changed when using our approach to generate counterfactuals. The experts select features that are more influential in determining the output of the model. On the other hand, our approach selects features that are more likely to change the outcome of a prediction. Our approach can play a key role in credit scoring for black-box model explanations and the credit scoring experts can leverage off the explanations from our approach, and this can result in a human-machine relationship.

D. MEAN OPINION SCORE (MOS)

The set of features that were used in the counterfactual explanations which were generated when using our approach, were compared to the set of features that were chosen by the credit scoring experts. Let L denote the number of credit scoring experts and I denote the number of features used to generate the counterfactuals when using our approach. We define below a_l^i which determines whether there is an overlap between a feature from our approach and the i^{th}-feature from the l^{th}-expert opinion,

$$a_l^i = \begin{cases} 1, & \text{if } e_l^i = m^i \\ 0, & \text{otherwise}, \end{cases} \tag{21}$$

where e_l^i is the l^{th}-expert feature opinion and m^i is the feature which is selected by our approach. Hence, the mean opinion score is given as follows,

$$MOS = \frac{1}{I}\sum_{i=1}^{I}\frac{1}{L}\sum_{l=1}^{L} a_l^i. \tag{22}$$

The MOS ranges between 0 and 1, values that are close to 1 would mean that the credit experts agree more with the features that need to be changed for the generation of a counterfactual using our approach. The values that are close

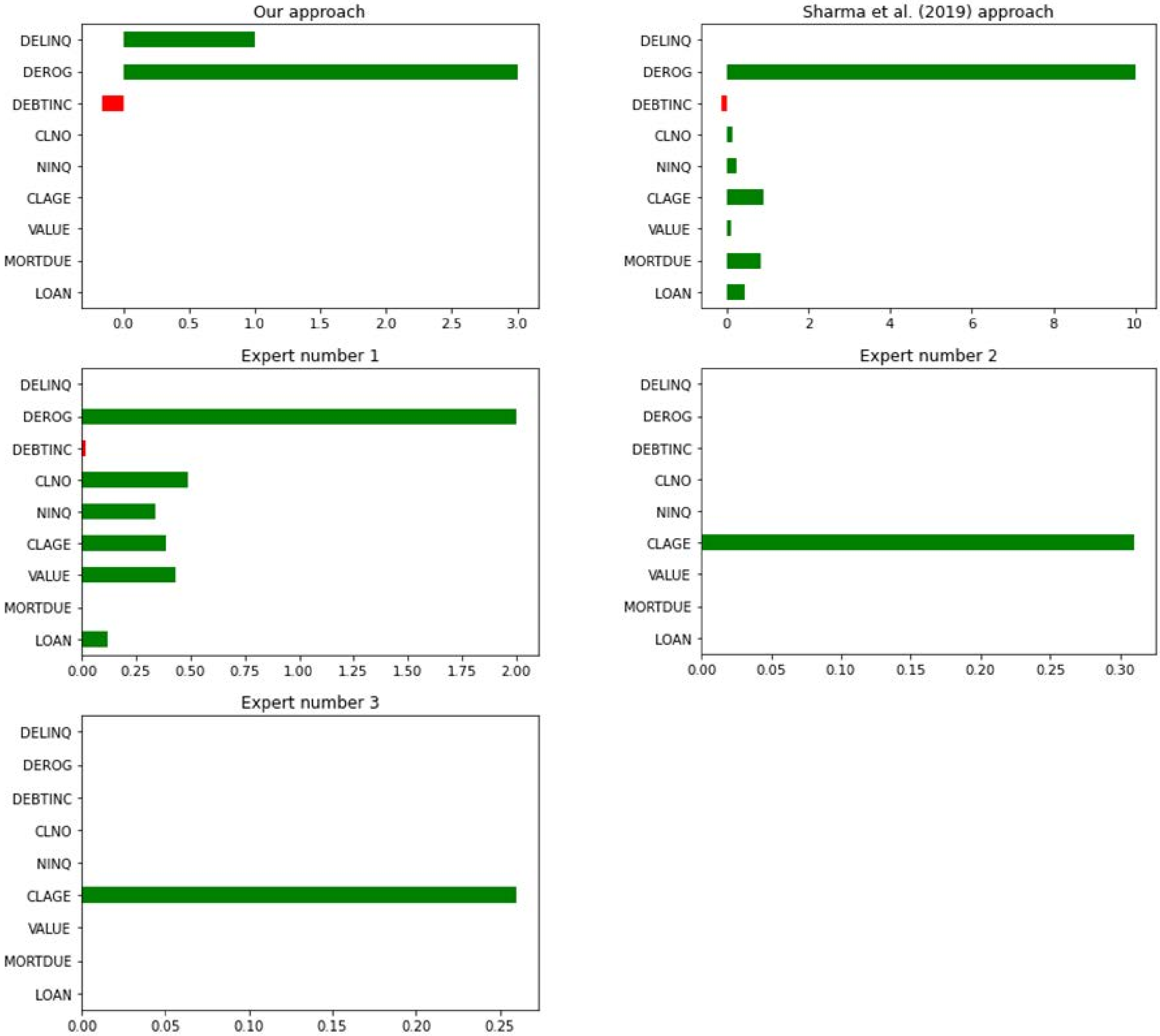

FIGURE 5. Explanation of on HMEQ credit dataset record using our approach, Sharma et al. [13] and experts. The red bars represent a decrease in a feature value and the green bars represent an increase in a feature value. The x-axis on each figure represent the change between the original feature value and the counterfactual feature value.

to 0, it would mean that the credit experts agree less with the values that are suggested for creating a counterfactual using our approach. For German credit scoring dataset, the $MOS = 0.17$, indicating that the credit experts agree 17% with the features that needed to be changed to generate a counterfactual. For HMEQ credit dataset, the $MOS = 0.083$, this indicates that the credit experts agree 8.30% with the features that needed to be changed to generate a counterfactual. The low values of the MOS are due to the fact that our approach looks at sparse counterfactuals, this results in few features that need to be changed, whereas for credit scoring experts there is no restriction in the number of required features.

E. COMPARISON WITH STATE-OF-THE-ART COUNTERFACTUAL EXPLANATION METHODS

Table 4 shows comparisons with state-of-the-art counterfactual explanation methods. Factors such as the black-box nature of the models, the applicability of the explanation approach using different classes of models (i.e. model-agnostic behaviour), involvement of domain experts, the sparseness of the counterfactuals, and quantitative assessments of the resulting counterfactuals, were used for comparison purposes. We observed that most methods are focusing on black-box models and model-agnostic behaviour of the explanations, except [3], [16], [30]. Our approach looks at all the suggested factors. The factors such as expert domain

TABLE 4. Comparison with state-of-the-art counterfactual methods with our approach. Legend: Q-Measured is Quantitatively Measured.

Source	Year	Black-box	Model-Agnostic	Experts	Sparse	Q-Measured
[11]	2018	✓	✓			
[3]	2018	✓	✓		✓	
[13]	2019	✓	✓			
[16]	2019	✓	✓		✓	✓
[31]	2019	✓	✓			
[32]	2019	✓	✓			✓
[12]	2019	✓	✓			
[33]	2020	✓	✓			
[15]	2020	✓	✓			
[30]	2020	✓	✓	✓	✓	
Our approach	2022	✓	✓	✓	✓	✓

knowledge, the sparseness of counterfactual explanations, and also the ability to quantitatively assess the generated counterfactuals, ensure the robustness of the generated counterfactual explanations. Please note that the list of sources in Table 4 is not exhaustive.

VI. CONCLUSION AND FUTURE WORK

The non-transparent nature of machine learning and deep learning models hampers the application of these models in credit scoring. We address this challenge of non-transparency by generating counterfactuals via a custom genetic algorithm to explain model predictions. We select predictive features and determine Spearman's rank correlations between the target flag and the predictors to enforce sparseness. We normalize all continuous features to expedite the generation of counterfactuals. We show the efficacy of the proposed approach on German and HMEQ credit scoring datasets. The experimental results indicate that the proposed approach efficiently generates sparse counterfactuals compared to similar methods. We also test the accuracy of our approach by using counterfactuals from credit scoring experts. Our results show an overlap between some features selected by the credit scoring experts and our approach in creating the counterfactuals.

Although the proposed method produces satisfactory results with the default parameter settings of the genetic algorithm, optimal parameter settings may improve the performance of the counterfactual generation. Furthermore, an optimal fitness function capturing different properties of counterfactuals using a genetic programming approach can enhance the performance of the proposed method. In addition, explaining the overall working mechanism of the black-box models instead of explaining individual instances can further improve the transparency and explainability of the credit scoring models.

ACKNOWLEDGMENT

The authors would like to thank the following credit risk experts for providing their subject matter expertise, Dr. Luyanda Ndlovu (Senior Credit Risk Specialist, South African Reserve Bank), Esinako Makaluza (Senior Credit Risk Specialist, South African Reserve Bank) and Lunga Dludla (Senior Manager, Credit Risk, African Bank), and also would like to thank the anonymous reviewers for providing valuable feedback on initial versions of this paper.

REFERENCES

[1] *Basel II: International Convergence of Capital Measurement and Capital Standards: A Revised Framework—Comprehensive Version, Bank for International Settlements*, BIS, Basel Committee Banking Supervision, Basel, Switzerland, 2006.

[2] P. Voigt and A. V. D. Bussche, *The EU General Data Protection Regulation (GDPR): A Practical Guide*, 1st ed. Cham, Switzerland: Springer, 2017.

[3] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the GDPR," *Harvard J. Law Technol.*, vol. 31, no. 2, pp. 87–841, 2018.

[4] A. E. Khandani, A. J. Kim, and A. W. Lo, "Consumer credit-risk models via machine-learning algorithms," *J. Bank Financ.*, vol. 34, no. 11, pp. 2767–2787, Nov. 2010.

[5] C. Molnar, *Interpretable Machine Learning*. 2019. [Online]. Available: https://christophm.github.io/interpretable-ml-book/

[6] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you?: Explaining the predictions of any classifier," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, pp. 1135–1144, 2016.

[7] H. Yang, C. Rudin, and M. Seltzer, "Scalable Bayesian rule lists," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 3921–3930.

[8] O. Gomez, S. Holter, J. Yuan, and E. Bertini, "Vice: Visual counterfactual explanations for machine learning models," in *Proc. 25th Int. Conf. Intell. User Interfaces*, 2020, pp. 531–535.

[9] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Mach. Intell.*, vol. 1, pp. 206–215, May 2019.

[10] S. Verma, J. P. Dickerson, and K. Hines, "Counterfactual explanations for machine learning: A review," 2020, *arXiv:2010.10596*.

[11] R. Mc Grath, L. Costabello, C. Le Van, P. Sweeney, F. Kamiab, Z. Shen, and F. Lecue, "Interpretable credit application predictions with counterfactual explanations," 2018, *arXiv:1811.05245*.

[12] R. Guidotti, A. Monreale, F. Giannotti, D. Pedreschi, S. Ruggieri, and F. Turini, "Factual and counterfactual explanations for black box decision making," *IEEE Intell. Syst.*, vol. 34, no. 6, pp. 14–23, Nov. 2019.

[13] S. Sharma, J. Henderson, and J. Ghosh, "CERTIFAI: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models," 2019, *arXiv:1905.07857*.

[14] A. Van Looveren, J. Klaise, G. Vacanti, and O. Cobb, "Conditional generative models for counterfactual explanations," 2021, *arXiv:2101.10123*.

[15] R. Poyiadzi, K. Sokol, R. Santos-Rodriguez, T. De Bie, and P. Flach, "FACE: Feasible and actionable counterfactual explanations," in *Proc. AAAI/ACM Conf. AI, Ethics, Soc.*, Feb. 2020, pp. 344–350.

[16] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proc. Conf. Fairness, Accountability, Transparency*, Jan. 2020, pp. 607–617.

[17] A. Artelt and B. Hammer, "Efficient computation of counterfactual explanations of LVQ models," 2019, *arXiv:1908.00735*.

[18] A. Van Looveren and J. Klaise, "Interpretable counterfactual explanations guided by prototypes," 2019, *arXiv:1907.02584*.

[19] K. Sokol and P. A. Flach, "Counterfactual explanations of machine learning predictions: Opportunities and challenges for ai safety," in *Proc. SafeAI@ AAAI*, 2019, pp. 1–4.

[20] M. Mitchell, *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1996.

[21] A. Hassanat, K. Almohammadi, E. Alkafaween, E. Abunawas, A. Hammouri, and V. B. S. Prasath, "Choosing mutation and crossover ratios for genetic algorithms—A review with a new dynamic approach," *Information*, vol. 10, no. 12, p. 390, Dec. 2019.

[22] D. Dua and C. Graff. (2017). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[23] (2017). Kaggle. *Home Loan Equity Data*. Accessed: Jun. 17, 2021. [Online]. Available: https://www.kaggle.com/ajay1735/hmeq-data

[24] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. 14th Int. Joint Conf. Artif. Intell.*, vol. 2, Aug. 1995, pp. 1137–1143.

[25] X. Dastile, T. Celik, and M. Potsane, "Statistical and machine learning models in credit scoring: A systematic literature survey," *Appl. Soft Comput.*, vol. 91, Jun. 2020, Art. no. 106263.

[26] D. West, "Neural network credit scoring models," *Comput. Oper. Res.*, vol. 27, no. 11, pp. 1131–1152, Sep. 2000.

[27] S. Ha and H.-N. Nguyen, "Credit scoring with a feature selection approach based deep learning," in *Proc. MATEC Web Conf.*, vol. 54, Dec. 2016, p. 05004.

[28] Y. Xia, C. Liu, Y. Li, and N. Liu, "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring," *Expert Syst. Appl.*, vol. 78, pp. 225–241, Jul. 2017.

[29] L. Munkhdalai, O.-E. Namsrai, and K. Ryu, "Credit scoring with deep learning," in *Proc. 4th Int. Conf. Inf., Syst. Converg. Appl.*, 2018, pp. 1–6.

[30] F. Cheng, Y. Ming, and H. Qu, "DECE: Decision explorer with counterfactual explanations for machine learning models," *IEEE Trans. Vis. Comput. Graphics*, vol. 27, no. 2, pp. 1438–1447, Feb. 2020.

[31] D. Mahajan, C. Tan, and A. Sharma, "Preserving causal constraints in counterfactual explanations for machine learning classifiers," 2019, *arXiv:1912.03277*.

[32] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee, "Counterfactual visual explanations," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 2376–2384.

[33] P. Wang and N. Vasconcelos, "SCOUT: Self-aware discriminant counterfactual explanations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8981–8990.

XOLANI DASTILE received the bachelor's degree majored in mathematics and mathematical statistics and the M.Sc. degree in mathematical statistics from Rhodes University, South Africa. He is currently pursuing the Ph.D. degree in computer science in the field of deep learning with the University of the Witwatersrand, Johannesburg, South Africa. He is also a Senior Data Scientist in a fuel management company, South Africa. Previously, he worked in major banks in South Africa under credit risk departments as a Quantitative Analyst and a Data Scientist.

TURGAY CELIK received the Ph.D. degree from the University of Warwick, Coventry, U.K., in 2011. He is currently a Professor in digital transformation and the Director of the Wits Institute of Data Science, University of Witwatersrand, Johannesburg, South Africa. His research interests include the areas of computer vision, (explainable) artificial intelligence, (health) data science, data-driven optimal control, and remote sensing. He is an Associate Editor of *BMC Medical Informatics and Decision Making*, *IET ELL*, IEEE Access, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, and *SIVP* (Springer).

HANS VANDIERENDONCK (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees from Ghent University, Belgium, in 1999 and 2004, respectively. He is currently a Professor in high-performance and data-intensive computing with the School of Electronics, Electrical Engineering and Computer Science, and the Director of the Centre for Data Science and Scalable Computing, Institute on Electronics, Communications and Information Technology, Queen's University Belfast, Belfast, U.K. His research aims to build efficient and scalable computing systems for data-intensive applications.