Model-agnostic Counterfactual Explanations in Credit Scoring

Dastile, X., Celik, T., & Vandierendonck, H. (2022). Model-agnostic Counterfactual Explanations in Credit Scoring. *IEEE Access*. Advance online publication. https://doi.org/10.1109/access.2022.3177783

Published in:
IEEE Access

Document Version:
Publisher's PDF, also known as Version of record

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

Publisher rights
© 2022 The Authors. This is an open access article published under a Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution and reproduction in any medium, provided the author and source are cited.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Open Access
This research has been made openly available by Queen's academics and its Open Research team. We would love to hear how access to this research benefits you. – Share your feedback with us: http://go.qub.ac.uk/oa-feedback
Model-agnostic Counterfactual Explanations in Credit Scoring

XOLANI DASTILE¹, TURGAY CELIK²,³,⁴, AND HANS VANDIERENDONCK⁵(Senior Member, IEEE)

¹School of Computer Science and Applied Mathematics, University of the Witwatersrand, Johannesburg 2000, South Africa.
²School of Electrical and Information Engineering, University of the Witwatersrand, Johannesburg 2000, South Africa.
³Wits Institute of Data Science, University of the Witwatersrand, Johannesburg 2000, South Africa.
⁴School of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, China
⁵School of Electronics, Electrical Engineering and Computer Science, Queen’s University Belfast, Belfast BT9 5BN, UK

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

ABSTRACT The past decade has shown a surge in the use and application of machine learning and deep learning models across different domains. One such domain is the credit scoring domain, where applicants are scored to assess their credit-worthiness for loan applications. During the scoring process, it is key to assure that there are no biases or discriminations that are incurred. Despite the proliferation of machine learning and deep learning models (referred to as black-box models in the literature) in credit scoring, there is still a need to explain how each prediction is made by the black-box models. Most of the machine learning and deep learning models are likely to be prone to unintended bias and discrimination that may occur in the datasets. To avoid the element of model bias and discrimination, it is imperative to explain each prediction during the scoring process. Our study proposes a novel optimisation formulation that generates a sparse counterfactual via a custom genetic algorithm to explain a black-box model’s prediction. This study uses publicly available credit scoring datasets. Furthermore, we validated the generated counterfactual explanations by comparing them to the counterfactual explanations from credit scoring experts. The proposed explanation technique does not only explains rejected applications, it can also be used to explain approved loans.

INDEX TERMS Credit Scoring, Machine Learning, Deep Learning, Counterfactual Explanation, Explainable AI, Genetic Algorithm.

I. INTRODUCTION

The past decade has shown a surge in the use and application of machine learning and deep learning models across different domains. One such domain is the credit scoring domain, where applicants are scored to assess their credit-worthiness for loan applications. During the scoring process, it is key to assure that there are no biases or discriminations that are incurred. Despite the proliferation of machine learning and deep learning models in credit scoring, there is still a need to explain how each prediction is made by the models. Most of the machine learning and deep learning models act as black-box models, hence these models are likely to be susceptible to unintended bias and discrimination that may occur in the datasets. To avoid the element of unintended model bias and discrimination, it is important to explain each prediction during the scoring process. The Basel Accord requires explanations for denied loan applications in order to assure that transparency is maintained in automated decisions in financial sectors. Emerging regulations such as the European Union General Data Protection Regulation (GDPR) stipulates that, for automated decisions, a “right to explanation” has to be maintained. Prediction explanations are crucial for high-stake decisions such as the presence or absence of a disease in the healthcare sector, rejection or acceptance of a loan application in the finance sector and denial or approval of a parole in the criminal justice sector.

The literature has done a great amount of work on explainable artificial intelligence (XAI) which is an emerging branch of artificial intelligence. 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 decision trees to explain the behaviour of a
black-box model (e.g. Local Interpretable Model-Agnostic Explanation (LIME) \cite{lime}), an ante-hoc explanation is an inherently interpretable model (e.g. Bayesian Rule List \cite{rudin}) and an instance-based explanation uses an instance to explain the behaviour of a black-box model (e.g. Visual Counterfactual Explanations (ViCE) \cite{vice}). Rudin \cite{rudin} 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, our study focuses on using a counterfactual explanation technique, which falls under the instance-based explanation. Wachter et al. \cite{counterfactual} highlighted three key benefits for using counterfactual explanations, (1) to inform and assist applicants to understand 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 classical example of a counterfactual explanation is a loan application \cite{counterfactual}: “Imagine you applied for a credit 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$ per month and if you would not have a second credit card.”

Despite the fact that 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 \cite{rashomon}. 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. Verma et al. \cite{counterfactual2} conducted a review on counterfactual explanations in machine learning applications. The study highlighted metrics for evaluation of generated counterfactuals, amongst other things. The counterfactual generation time was a metric discussed amongst other metrics. This metric can be taken as an average over the generation of a counterfactual for a group of records or for the generation of multiple counterfactuals for a single input record Verma et al. \cite{counterfactual2}.

Our study tries to address these issues and our contributions are as follows: 1) a novel formulation of the optimisation problem that leads to a single sparse counterfactual explanation using a custom genetic algorithm; 2) a better generation time for counterfactuals that is accounted by the normalisation of continuous features and the selection of predictive features; and 3) validating counterfactual explanations using credit scoring experts.

Without loss of generality, we used counterfactual and counterfactual explanation interchangeably. Further, the following words; sample, instance and record, are used interchangeably.

The remainder of this paper is organised as follows. Section \ref{related-work} discusses related work on counterfactual explanations. Section \ref{methodology} introduces our proposed methodology for generating counterfactual explanations. The experiment setup is described in Section \ref{experiment} and the results are given in Section \ref{results}. Finally in Section \ref{discussion}, we summarise the paper and discuss our future work.

\section{Related Work} \label{related-work}
In recent years, research has shown an increase in the number of studies that focused on explaining machine learning predictions using counterfactuals. Grath et al. \cite{counterfactual3} 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 weights that are produced from feature importance produce more compact counterfactuals. Furthermore, the study produced positive counterfactuals for accepted loans to assist individuals when they make future financial decisions.

In most cases, counterfactuals are derived by using a metaheuristic approach (i.e. an optimisation approach that is not problem/task dependent). Guidotti et al. \cite{counterfactual4} proposed a method that learns a local and interpretable classifier on a synthetic neighbourhood of a record of interest (i.e. an instance that needs its prediction 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 which 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. \cite{counterfactual5} proposed a unified and model agnostic approach to address non-transparency (amongst 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. \cite{counterfactual5} 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 small number of features makes counterfactuals to be feasible. Van Looveren et al. \cite{counterfactual6} proposed a framework for generating sparse and in-distribution counterfactuals. A sparse counterfactual refers to a counterfactual that requires minimum feature changes to make the counterfactual to belong to a desired class.

Poyiadzi et al. \cite{counterfactual7} 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, and this makes counterfactuals to be feasible and actionable. Mothilal et al. [16] posited that the counterfactuals should be feasible and diversified. The study proposed a framework for generating and assessing diversity of counterfactuals based on a determinant of a kernel matrix. The proposed framework generate diverse counterfactuals as opposed to previous methods.

Efficiency and speed are the key factors when generating counterfactuals. It is better to use less computer resources to speed up the computation time. Methods that are resource intensive, 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.

Not only speed and efficiency are key when generating counterfactuals, but also 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 safety and security of the system and the possibility of data leakage. This pose 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.

Most of the methods mentioned here generate more than one counterfactual. The problem with multiple counterfactuals is that they might have contradicting paths because of the diversification. The other issue with counterfactuals is the time it takes to generate them, which might be impacted by the number of features in a dataset and also a huge varying degree of within variance in the features. In our study, we attempt to address these issues.

III. METHODOLOGY

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

Figure 1 shows the flowchart of the methodology. Firstly, we focus on feature selection, where we select predictive features using a random forest model. Secondly, we calculate the 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 we generate 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 normalised. The purpose of normalising continuous features is to assure 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 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 assure 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.

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

$$I_E = - \sum_{j=0}^{1} p_j \log_2 p_j \quad (1)$$

or the gini index

$$I_G = 1 - \sum_{j=0}^{1} p_j \quad (2)$$

FIGURE 1. Methodology flowchart.
where $p_j$ represents proportion of samples in each class. The more “pure” a feature is, the more predictive it is.

### B. COUNTERFACTUAL SPARSITY

To assure that the counterfactuals that are generated are sparse, we used a Spearman’s rank correlation coefficient. The Spearman’s correlation 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). The Spearman’s rank correlation is given as

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

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.

### C. NORMALISATION OF CONTINUOUS FEATURES

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

$$
\mathbf{f'}_{i, \text{norm}} = \frac{f_i - \min(f_i)}{\max(f_i) - \min(f_i)},
$$

such that

$$
\mathbf{f'}_{i, \text{norm}} \in [0, 1],
$$

where $\min(f_i)$ and $\max(f_i)$ are the minimum and the maximum values for feature $i$, respectively.

### D. GENETIC ALGORITHMS

The genetic algorithm is a heuristic search approach that is motivated by the natural evolution \[20\]. The genetic algorithm searches for optimal values that can either minimise or maximise a certain function. Hence, genetic algorithms are used for optimisation problems. The genetic algorithm process 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 the characteristics of the parents. The offspring then gets included in the next generation of the population. The process repeats itself until it converges to a solution. The genetic algorithm process involves five phases i.e. initial population, fitness function, selection, cross-over and mutation. In the context of credit scoring, a population individual is a feature vector and this feature vector is regarded as a counte-factual. To select the best counterfactuals, a fitness function is used. The fitness function is the black-box model that will be explained by the counterfactual. The fitness function output is binary (i.e. default or non-default). The quality of each counterfactual is determined by the output of the fitness function being non-default and the distance between the counterfactual and the feature vector of interest (i.e. a feature vector that we want to explain its prediction) should be less than some epsilon value. Selected counterfactuals go to a mating pool and these counterfactuals are known as the parents. Every two parents will produce two offspring (i.e. two counterfactuals). The mating process is actually 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 on the offspring in order to generate new offspring. The changes are known as the mutation phase. 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 OPTIMISATION PROBLEM

This section defines our formulation for the optimisation problem which is solved via a custom genetic algorithm in order to generate counterfactuals. Let $c^* \in \mathbb{R}^d$ denote a counterfactual for record $x \in \mathbb{R}^d$ and $g$ denote a black-box model. The black-box model is given as

$$
g : \mathbb{R}^d \rightarrow \{0, 1\},
$$

and the aim is to find $c^* \in \mathbb{R}^d$ such that

$$
g(c^*) = \min_{c \in \mathbb{R}^d} g(c)
$$

and subject to

$$
dist(c^*, x) < \epsilon
$$

and

$$
c^* \in [\min(f_i), \max(f_i)], \forall i \in \{1, 2, \cdots, d\}.
$$

The main idea behind the above optimisation problem is to find optimal values $c^* \in \mathbb{C}$ that result in a non-default class $g(c^*) = 0$, ensuring that the generated counterfactual $c^*$ is as close as possible to the instance of interest $x$ and that $c^*$ comes from the same data distribution as $x$. According to \[7\], closeness should be measured by the $L_1$ norm distance divided by the mean absolute deviation (MAD). Wachtler et al. \[7\] 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 (10) 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

\[
\text{dist}(c^*, x) = \frac{d^*}{MAD^*} + \frac{d}{MD^*},
\]

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

\[
MAD^i = \frac{1}{n} \sum_{j=1}^{n} |f_j^i - \bar{f}^{(i)}|,
\]

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

\[
MD^i = \frac{1}{m} \sum_{l=1}^{m} o_l |f_l^i - \tilde{f}^{(i)}|,
\]

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

The threshold for the chosen distance is defined by \(\epsilon\) which is provided by the user. The fitness function in our optimisation problem is \(g(c)\). To explain predictions of applicants that are approved for loans, the aim is to find optimal values of \(c^*\) such that

\[
g(c^*) = \max_{c \in R^d} g(c)
\]

subject to the same constraints of Equation (7). The major differences between our study and that of Sharma et al. [13] are 1) the formulation of the optimisation problem, 2) the distance calculation of categorical features and 3) the generation of sparse counterfactuals. Sharma et al. [13] used the following fitness function,

\[
\text{fitness function} = \frac{1}{\text{dist}(c^*, x)}
\]

where

\[
\text{dist}(c^*, x) = \frac{n_{\text{con}}}{n} \sum_{i=1}^{n_{\text{con}}} \frac{|c^*_i - f_i^*|}{MAD^i} + \frac{n_{\text{cat}}}{n} \text{SimpleMat}(c^*_\text{cat}, x_{\text{cat}})
\]

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

\[
\text{SimpleMat} = \frac{\text{number of matching attributes}}{\text{total number of attributes}},
\]

and has values between 0 and 1.

IV. EXPERIMENTS

A. DATA

Publicly available credit scoring datasets were considered 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) or “non-default” (i.e. good applicants).

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 \(D_1, D_2, \ldots, D_K\), where

\[
\bigcup_{k=1}^{K} D_k = X.
\]

Each fold \(D_k\) is used as a test set and the remaining \(D = \{D_j\}_{j=1}^{K-1} \setminus \{D_k\}\) folds are used for model training. The overall performance metric for the model is

\[
g_p = \frac{1}{K} \sum_{k=1}^{K} g^k_p
\]

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

C. EXPERIMENT SETUP

For genetic algorithm, we ran four experiments on each dataset to select optimal parameters based on the execution times for the optimisation problem, the resulting objective function and the distance between the record that needs to be explained and the generated counterfactual. Table I shows different parameter values for our experiments for the genetic algorithm and the times (measured in seconds) to execute the optimisation problem , the resulting objective functions and the distances are also shown. For German datasets, 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 objective function
output (i.e. the non-default output), and the minimal distance. For HMEQ datasets, we chose $\epsilon = 7$, mutation rate = 0.04, cross-over rate = 0.70 and population size = 100, since they are giving the desired objective function 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 network (this was to assure 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, maximum depth for each decision tree was set to 5, the number of decision trees was set to 100. For the artificial neural network, we used 3 hidden layers, the first hidden layer had 50 neurons, 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 network 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
Three experts from financial institutions were willing to partake in the experiments of this study. All of these experts have more than 7 years working experience in the financial sector as quantitative analysts in credit scoring space. They all have a background in developing credit scorecards from scratch. The experts were given a data dictionary with a 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 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 credit scoring prediction. The credit risk experts then created their own counterfactuals and they assessed the prediction of their counterfactuals using an online user interface. The experts were also asked to normalise 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.

V. RESULTS
This section is for the results that were obtained from our experiments. 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 network classifier on HMEQ credit dataset are assessed. 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 was 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.

B. MODEL PERFORMANCE
Table 2 shows the classifier performance results that were reported in literature, and we compared those results to the performance of the models that were used in our study. 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 made a certain decision and what actions are required to effect a desired outcome for the loan applicant.

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

| Experiment | German | HMEQ |
|------------|--------|------|
|           | $\epsilon = 4$ | $\epsilon = 4$ |
| Experiment 1 | mutation rate = 0.01 | mutation rate = 0.01 |
| | cross-over = 0.40 | cross-over = 0.40 |
| | population = 30 | population = 30 |
| | time = 87, func = 0, dist=15 | time = 87, func = 1, dist=15 |
| Experiment 2 | 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=24 | time = 87, func = 1, dist=15 |
| Experiment 3 | 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 = 125, func = 1, dist=195 |
| Experiment 4 | 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 |

TABLE 2. Literature results relating to performances of credit scoring models.

**Source** | **Year** | **Model** | **German** | **HMEQ** | **Performance metrics**
|-----------|----------|-----------|-----------|-----------|-------------------|
| 26        | 2000     | Logistic Regression | 0.76 | - | time=7.97, func=0, dist=5 |
| 27        | 2016     | Neural Network | 0.75 | - | time=7.72, func=0, dist=16 |
| 28        | 2017     | XGBoost | 0.77 | - | Our approach |
| 29        | 2018     | Neural Network | 0.78 | 0.80 | time=386, func=0, dist=3.75 |
| **Our Study** | 2022 | Random Forest | 0.74 | 0.76 | time=192, func=0, dist=6 |
| **Our Study** | 2022 | Neural Network | - | - | - |

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 dataset, respectively. Please note that all continuous values in the figures were normalised, however when providing explanations using text, the normalised values were denormalised 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 [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 [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.

The difference between our approach and that of [13] is around the time it takes to generate a counterfactual and the distance between the record of interest and the generated counterfactual. Since our approach is using a binary fitness function, the process of convergence is much quicker compared to the approach presented in [13], and also 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 experts generated their own counterfactual explanations based on their knowledge domain. Please refer to Figure 4 and Figure 5 to visually see the explanations that are given in the text below. Expert number 1 on the 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 and Age (years) increases by 7. Expert number 2 on the 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 7. Expert number 3 on the German credit dataset, stated that the applicant would qualify for the loan if the Credit Amount increases by 8542, Age (years) decreases by 10 and Payment Status of Previous Credit decreases by 3. Expert number 1 on the HMEQ credit dataset, stated that the applicant would qualify for the loan if the LOAN increases by 11544, VALUE increases by 373079, CLAGE increases by 456, NINQ increases by 8, CLNO increases by 42 and DEBTINC increases by 24. Expert number 2 on the HMEQ credit dataset, stated that the applicant would qualify for the loan if the the CLAGE increases by 362. Expert number 3 on the 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 we used 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 \( \alpha^i \) which determines whether there is an overlap between the features from our approach and the \( i^{th} \) feature.
FIGURE 5. Explanation of on HMEQ credit dataset record using our approach, [13] and experts. The red bars represent a decrease in a feature value and the green bars represent an increase in a feature value.

from the $l^{th}$-expert opinion,

$$a_i^l = \begin{cases} 1, & \text{if } e_i^l = m^i \\ 0, & \text{otherwise} \end{cases} \quad (19)$$

where $e_i^l$ 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,

$$MOS = \frac{1}{L} \sum_{l=1}^{L} \frac{1}{L} \sum_{i=1}^{L} a_i^l. \quad (20)$$

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. The values that are close to 0, it would mean that the credit experts agree less with the values that are suggested for creating a counterfactual. For German credit scoring dataset, the $MOS = 0.17$, indicating that the credit experts agree 17% with the features that needed to be changed in order 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 in order to result into 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.
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, applicability of the approach using different classes of models (i.e., model-agnostic), involvement of domain experts, sparseness of the counterfactuals and quantitative assessments of the resulting counterfactuals, were used for comparison purposes. We observe that most methods are focusing on black-box models and model-agnostic behaviour of the explanations, with the exception of [7, 16, 30]. Our approach looks at all the suggested factors, and factors such as expert domain knowledge, sparseness of counterfactual explanations and also the ability to quantitatively assess the generated counterfactuals, assure robustness of the generated counterfactual explanations. Please note that the list of sources in Table 4 is not exhaustive.

VI. CONCLUSION AND FUTURE WORK

In this paper, we selected predictive features to expedite the generation of counterfactuals. The correlations between the selected features and the target variable were assessed for the purpose of generating sparse counterfactuals. All continuous features were then normalised to deal with high within feature variability. A novel optimisation problem was formulated to generate counterfactuals via a custom genetic algorithm. In general, explanations when using counterfactuals provide insights about the actions that are needed to be taken in order to get a desired outcome. In the case of credit scoring, a desired outcome is an approved loan application. The explanations from generated counterfactuals were compared to explanations that were provided by the credit scoring experts. The results showed an overlap between some of the features that needed to be changed when using the proposed approach and when using the credit risk experts’ opinions. In practice, credit scoring experts can leverage off the explanations that are generated by our approach to explain the predictions from black-box models. Further, our results showed that our approach takes into consideration important factors such as sparseness and quantitative assessment of generated counterfactuals, compared to the state-of-the-art counterfactual explanation methods. An immediate future work is the creation of a user interface (UI) of our approach. Also, the future work should focus on extending this work to explain the overall workings of the black-box models instead of looking at individual instances.

ACKNOWLEDGMENTS

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).

REFERENCES

[1] Basel Committee on Banking Supervision, “Basel ii: International convergence of capital measurement and capital standards: A revised framework - comprehensive version, bank for international settlements,” BIS, 2006.
[2] P. Voigt and A. v. d. Bussche, The EU General Data Protection Regulation (GDPR): A Practical Guide, 1st ed. Springer Publishing Company, Incorporated, 2017.
[3] M. T. Ribeiro, S. Singh, and C. Guestrin, “‘why should I trust you?’: Explaining the predictions of any classifier,” Knowledge Discovery in Databases, pp. 1135—1144, 2016.
[4] H. Yang, C. Rudin, and M. Seltzer, “Scalable bayesian rule lists,” in International Conference on Machine Learning, 2017, pp. 3921–3930.
[5] O. Gomez, S. Holter, J. Yuan, and E. Bertini, “Vice: visual counterfactual explanations for machine learning models,” in Proceedings of the 25th International Conference on Intelligent User Interfaces, 2020, pp. 531–535.
[6] C. Rudin, “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead,” Nature Machine Intelligence, vol. 1, no. 5, pp. 206–215, 2019.
[7] S. Wachter, B. Mittelstadt, and C. Russell, “Counterfactual explanations without opening the black box: Automated decisions and the gdpr,” Harvard Journal of Law and Technology, vol. 31, no. 2, p. 841–87, 2018.
[8] A. E. Khandani, A. J. Kim, and A. W. Lo, “Consumer credit-risk models via machine-learning algorithms,” Journal of Banking & Finance, vol. 34, no. 11, pp. 2767–2787, 2010.
[9] C. Molnar, Interpretable Machine Learning, 2019, https://christophm.github.io/interpretable-ml-book/.
[10] S. Verma, J. P. Dickerson, and K. Hines, “Counterfactual explanations for machine learning: A review,” CoRR, 2020.
[11] R. M. Grath, L. Costabello, C. L. Van, P. Sweeney, F. Kambia, Z. Shen, and F. Lecue, “Interpretable credit application predictions with counterfactual explanations,” arXiv preprint arXiv:1811.05245, 2018.
[12] R. Guidotti, A. Monreale, F. Giannotti, D. Pedreschi, S. Ruggieri, and F. Turini, “Factual and counterfactual

Table 4. Comparison with state-of-the-art counterfactual methods with our approach.

| Source | Year | Black-box | Model-Agnostic | Experts | Sparse | Quantitative |
|--------|------|-----------|----------------|---------|--------|--------------|
| Our approach | 2022 | ✓ | ✓ | ✓ | ✓ | ✓ |
accounts for black box decision making,” IEEE Intelligent Systems, vol. 34, no. 6, pp. 14–23, 2019.
[13] S. Sharma, J. Henderson, and J. Ghosh, “CertifiAI: Counterfactual explanations for robustness, transparency, interpretability, and fairness of artificial intelligence models,” arXiv preprint arXiv:1905.07857, 2019.
[14] A. Van Looveren, J. Klaise, G. Vacanti, and O. Cobb, “Conditional generative models for counterfactual explanations,” arXiv preprint arXiv:2101.10123, 2021.
[15] R. Poyiadzi, K. Sokol, R. Santos-Rodriguez, T. De Bie, and P. Flach, “Face: feasible and actionable counterfactual explanations,” in Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society, 2020, pp. 344–350.
[16] R. K. Mothilal, A. Sharma, and C. Tan, “Explaining machine learning classifiers through diverse counterfactual explanations,” in Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency, 2020, pp. 607–617.
[17] A. Artelt and B. Hammer, “Efficient computation of counterfactual explanations of lvq models,” arXiv preprint arXiv:1908.00735, 2019.
[18] A. Van Looveren and J. Klaise, “Interpretable counterfactual explanations guided by prototypes,” arXiv preprint arXiv:1907.02584, 2019.
[19] K. Sokol and P. A. Flach, “Counterfactual explanations of machine learning predictions: opportunities and challenges for ai safety.” in SafeAI@ AAAI, 2019.
[20] M. Mitchell, An Introduction to Genetic Algorithms. MIT Press, 1996.
[21] A. Hassanat, K. Almohammedi, E. Alkaafween, 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, 2019.
[22] D. Dua and C. Graff, “UCI machine learning repository,” 2017. [Online]. Available: http://archive.ics.uci.edu/ml
[23] Kaggle, Home Loan Equity Data, 2017 accessed 17 June 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 Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2, ser. IJCAI’95. Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143.
[25] X. Dastile, T. Celik, and M. Potsane, “Statistical and machine learning models in credit scoring: A systematic literature survey,” Applied Soft Computing, vol. 91, p. 106263, 2020.
[26] D. West, “Neural network credit scoring models,” Computers & Operations Research, vol. 27, no. 11, pp. 1131 – 1152, 2000.
[27] S. Ha and H.-N. Nguyen, “Credit scoring with a feature selection approach based deep learning.” MATEC Web
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 of high-performance and data-intensive computing with the School of Electronics, Electrical Engineering and Computer Science, and Director of the Centre for Data Science and Scalable Computing in the 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.

* * *