Counterfactual learning in customer churn prediction under class imbalance

Yuanyuan Li
Business School, Sichuan University
Chengdu, P.R. China
lyyjos@163.com

Xue Song
Business School, Sichuan University
Chengdu, P.R. China
2750883144@qq.com

Taicheng Wei
Guangxi Tobacco Industrial Co.,Ltd
Nanning, P.R. China
304125155@qq.com

Bing Zhu
Business School, Sichuan University
Chengdu, P.R. China
zhubing1866@hotmail.com

ABSTRACT

Nowadays, in churn prediction, many decision-makers are trying to obtain knowledge from models through interpretation techniques due to the non-transparency of black-box. In these techniques, the counterfactual explanation generated by the counterfactual learning method is an easy-to-understand and quantitative explanation of a single instance in black-box model prediction. Counterfactual learning relies on the transition of instances across the decision boundary, while the impact of class imbalance issue and instance position in customer-related data is insufficiently considered in recent studies. In this case, this research innovatively proposes that when generating counterfactual explanations, the impact of class imbalance issue and the instance location in customer data needs to be considered. And through comparative experiments we prove that there are obvious differences in the success rate of finding a counterfactual explanation, the distance between the counterfactual explanation and the original instance (i.e. proximity), the proportion of feature change (i.e. sparsity), and the degree of proximity support (i.e. credibility) with the original instance in different instance locations and unbalanced scenarios. In addition, in our experiments, the impact of class imbalance and instance positions vary among counterfactual methods. This research provides a reference for the application of counterfactual learning in the field of customer churn prediction and elaborates that when counterfactual learning is used, the influence of class imbalance and instance location needs to be considered.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning approaches**; • **Machine learning**; • **learning**;

KEYWORDS

counterfactual learning, model interpretation, class imbalance, churn prediction, instance position

ACM Reference Format:
Yuanyuan Li, Xue Song, Taicheng Wei, and Bing Zhu. 2023. Counterfactual learning in customer churn prediction under class imbalance. In *2023 6th International Conference on Big Data Technologies (ICBDT) (ICBDT 2023), September 22–24, 2023, Qingdao, China.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3627377.3627392

1 INTRODUCTION

Currently, many companies make churn predictions in their daily operation and apply black-box models such as deep learning and ensemble learning models in their prediction. However, due to the non-transparency of black-box models, the knowledge obtained by decision-makers from the prediction process is limited. Business executives are often curious about why a model makes certain predictions, but the "black-box models" make it difficult for them to understand how customer behavior features influence the model's predictions and why the model makes such predictions. As a result, they cannot trust the model's predictions and find it difficult to develop customer management strategies based on them [18].

To gain more insights from the prediction process, counterfactual learning methods are commonly used as one of the interpretation techniques for black-box models to interpret individual instances' predictions [9], especially in fields such as finance and healthcare. Counterfactual learning methods generate counterfactual explanations for individual instances by continuously changing certain feature values of the instances until a "leap" in classification occurs. A counterfactual explanation statement in the following form: "If X did not happen, Y would not happen." Due to its unique and straightforward way of explanation, counterfactual learning has been widely applied in the research of model interpretability. For example, DASTILE et al. generate sparse counterfactual explanations for explaining predictions of black-box models using a custom genetic algorithm. This method can not only explain rejected loan applications but also explain approved loan applications [5]. Carlos et al. demonstrated the superiority of counterfactual explanations over feature importance methods in data-driven decision-making, showing advantages in quantitative decision guidance [6].

With customer-related data growing rapidly through digital operation, the source data in churn prediction has become more complex,

ICBDT 2023, September 22–24, 2023, Qingdao, China

ACM ISBN 979-8-4007-0766-7/23/09…$15.00
https://doi.org/10.1145/3627377.3627392

for example, class imbalance is an issue that needs to be addressed. Class imbalance refers to a situation in binary classification tasks where the majority of the data belongs to one class (the majority class), while the other class is represented by a smaller amount of data (the minority class). For example, it is reported that the three-month retention rate of Facebook users is around 0.98, indicating that the number of churned customers (only 2 percent) is much lower than the retained customers (98 percent). And the 12-month retention rate of the Lyft platform is around 0.22, with churned customers (78 percent) outnumbering the retained customers (22 percent). The issue of class imbalance in churn prediction is particularly critical for products and services provided for businesses. When class imbalance exists, classifiers tend to exhibit poorer accuracy when predicting the minority class, thereby affecting the decision boundary of classification. Based on the working principle of counterfactual learning, the generation of counterfactual explanations depends on identifying the decision boundary of predictions, with instances located on the decision boundary being more likely to undergo classification transitions. Therefore, this research innovatively proposes that the effectiveness of counterfactual learning in generating individual instance explanations is influenced by the degree of class imbalance of original data and is closely related to the position of the instance. However, existing research on counterfactual learning has not considered the impact of class imbalance and instance position on the quality of counterfactual learning.

In this case, this paper will conduct experiments to study counterfactual explanation generation in conjunction with class imbalance. We will first select datasets with different degrees of class imbalance to construct black-box models. Then, the instances of churned users (minority class) predicted by the models will be labeled based on their positions. Next, different counterfactual methods will be chosen to compare the effects of class imbalance and instance position on counterfactual learning methods through comparative experiments. We will focus on the following research questions:

1) Does the degree of class imbalance affect the quality of counterfactual explanation generation?
2) Does the influence of class imbalance differ among instances in different positions within the same class (e.g., instances located on the decision boundary versus safe instances)?
3) Do different methods for generating counterfactual explanations result in differences in the generated explanations under class imbalance conditions?

The structure of this paper is as follows: Section 2 reviews the current literature. Section 3 presents the research methodology. Section 4 conducts experiments and analyzes the experimental results. Finally, Section 5 concludes the paper.

2 RELATED WORKS

Nowadays there are more and more researchers trying to understand the decision progress of models in customer churn prediction. For example, Hasumoto and Goto conducted a visual analysis of churn prediction models using the PDP technique. They explained the extracted features and the implications of potential customer purchase behavior within the potential churner group, providing support for effective retention strategies [8]. However, little attention has been paid to interpreting predictions from local vision.

In recent years, an increasing number of scholars have started to pay attention to counterfactual learning and consider counterfactual explanations as one of the most valuable methods among local post-hoc interpretation approaches. Counterfactual explanations were initially proposed by Wachter et al., who generated post-hoc instances to explain predictions of black-box models [17]. Currently, based on the methods for generating counterfactual explanations, they can be mainly categorized as follows:

Optimization-based methods: Optimization-based counterfactual explainers define a loss function that captures the desired attributes of an explanation and utilizes existing optimization algorithms to minimize the loss. Most counterfactual explainers solve optimization problems to generate counterfactuals [7]. For instance, Wachter et al. pioneered the use of gradient descent to query the best counterfactual instances [2]. Dastile and Dandl used genetic algorithms or multi-objective genetic algorithms [5][4] to generate counterfactual explanations more effectively.

Heuristic search-based methods: Heuristic search-based counterfactual explanation methods aim to discover counterfactuals through local and heuristic selection, minimizing a cost function at each iteration. They generally exhibit higher efficiency compared to optimization algorithms. For example, Martens and Provost proposed a model-agnostic heuristic explanation method called SEDC, which guides local improvements through best-first search and pruning [13]. Laugel et al. introduced the GSG method, which relies on partitioning a range of synthetic instances around the instance to be explained to find the closest counterfactual explanation [12].

Instance-based methods: Instance-based counterfactual explanations retrieve counterfactuals by selecting the most similar examples from the dataset. For instance, the Nearest Neighbor Counterfactual Explanation (NNCE) method proposed by Shakhnarovich et al. selects instances that are most similar to the instance to be explained but have different labels as candidate counterfactuals, ranking them based on their similarity to the instance. The method then selects the top-k most similar counterfactual explanations [15]. The Neighbor-based Instance Counterfactual Explanation (NICE) method proposed by Brughmans and Martens accelerates the search process by iteratively introducing feature values from neighbors that differ the most, using information from the nearest distinct neighbors [1]. Poyiadzi et al. proposed the FACE method, which identifies feasible paths through density analysis to generate more feasible counterfactual explanations [14].

Hybrid methods: Mothila et al. combined the LIME and SHAP feature attribution methods with counterfactual explanations. They proposed an explanation method that reveals actual causal relationships uncovered by counterfactual explanations and used both attribution methods to explain key outcomes of the model [10].

Based on the existing research, there is a lack of counterfactual explanation methods that consider class imbalance. Data in customer churn prediction are typically class-imbalanced, which directly affects the effectiveness of counterfactual explanations for minority class instances. Regarding interpretability research under class imbalance conditions, only Dablain et al. have proposed a method for understanding deep learning models from the perspectives of class prototypes, sub-concepts, and outlier instances. This method utilizes imbalance learning algorithms to detect important features and class distribution prototypes that are crucial

to model performance [3]. However, this research solely discusses the explanation of deep neural networks from a prior perspective and has not yet explored the consideration of class imbalance in post-hoc interpretability. Further research is needed to investigate how to improve the interpretability performance of counterfactual methods under class imbalance conditions.

3 CLASS IMBALANCE IN COUNTERFACTUAL LEARNING

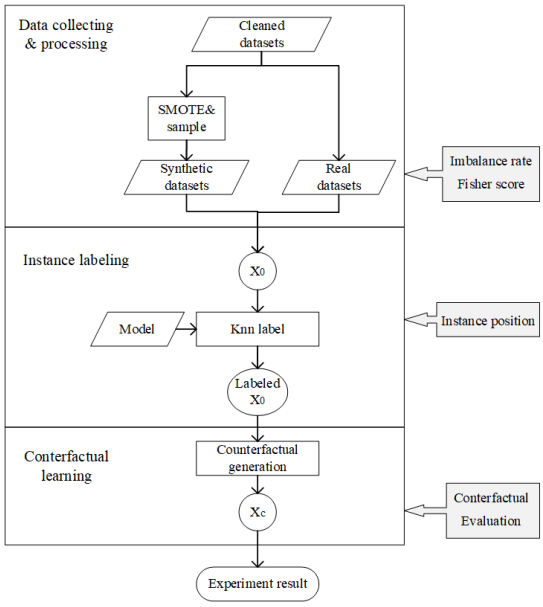

Figure 1: Research workflow

In this section, we will first define the research problem of counterfactual learning. Then, the detailed research steps will be elaborated, and the experimental variables, class imbalance, and instance position will be presented.

The problem of counterfactual learning in binary classification can be described in the following way: Given a classifier b that outputs the decision $Y = b(X_0)$ for an instance X_0, a counterfactual explanation of X_0 consists of an instance X_c such that the decision for b on X_c is different from Y, i.e., $b(X_c) = Y_c$, where Y_c is the opposite class of Y. A good counterfactual explanation is such that the difference between X_0 and X_c is minimal.

The research process consists of the following steps according to Figure 1: 1)Data Collection and Class Imbalance Control: In this step, data will be collected, and measures will be taken to control the class imbalance within the dataset. This will ensure the availability of a suitable dataset for experimentation. 2)Instance Position Labeling: A black-box model b will be built in this stage. Using this model, a prediction $Y = b(X_0)$ will be generated for the dataset obtained in the previous step. Then a KNN labeling progress will label X_0 based on its position against the decision border. This partitioning will distinguish instances located at the decision boundary from safe instances, enabling a more refined analysis. 3)Counterfactual Learning: In this stage, we use different counterfactual learning methods to find the counterfactual explanations of the labeled X_0. Then, the obtained counterfactual explanations X_c will be compared with the original X_0 under the criterion of counterfactual evaluations. The experimental results, obtained through this analysis, will provide information regarding the influence of class imbalance, instance position, and the effectiveness of different counterfactual explanation generation methods.

3.1 Counterfactual learning methods

To compare the differences in different counterfactual learning methods under the condition of class imbalance, we select three different kinds of counterfactual learning methods. They are Wachter's method, Growing Spheres(GS), and Actionable Recourse(AR).

3.1.1 Wachter's method. Wachter et al. first used counterfactual learning as a machine learning explanation method in 2017 and proposed an optimization-based counterfactual explanation generation algorithm, which is one of the current classical counterfactual algorithms [17]. Wachter et al. suggest minimizing the following loss function:

$$L(x, x', y', \lambda) = \lambda(\hat{f}(x') - y')^2 + d(x, x') \tag{1}$$

This loss function measures the difference between the counterfactual prediction and the predefined outcome and the gap between the counterfactual and the instance of interest. When the instance x to interpret is selected, the method randomly chooses an instance as the initial counterfactual situation and takes it as the starting point to optimize the loss function. By increasing λ, and taking the current counterfactual situation as a starting point, the method optimizes the loss function, and returns the counterfactual situation that minimizes the loss. Then the method repeats this progress and returns a list of counterfactual situations or a counterfactual situation that minimizes losses.

3.1.2 Growing Spheres. The Growing Spheres (GS) algorithm, proposed by Laugel et al. in 2017, is a random-search-based method that generates counterfactual explanations by determining the minimum change required to change the prediction: given the data points whose classification must be explained, the proposed method consists in identifying neighbors of different classifications, where the nearest neighbor definition integrates sparsity constraints [11]. The GS method is proposed from the point where end-user access to model knowledge is scarce. In this case, the need for comparison-based interpretability tools, which do not rely on any prior knowledge, including any existing data, constitutes one of the main motivations of the GS algorithm. GS method contains two steps. Firstly, the GS method generates observations in the feature space of the L2 spherical layer around the initial instance x until different class instances are found. Secondly, let e be the closest instance of a different class found in the first step. GS then uses a naive heuristic method that is based on the least coordinates of $e - x$ and may have a low local correlation with the classifier decision boundary.

3.1.3 Actionable Recourse. Actionable Recourse (AR) was proposed by USTUN et al as a machine learning decision interpretation method based on the linear classification in 2019, which ensures that linear classification decision interpretation problems are solved

without interfering with model development through integer programming tools [16]. The goal of the AR method is defined as that given an instance is assigned an undesired result $f(x) = -1$, the goal of AR is to find an action a by solving an optimization problem in a form such that $f(x + a) = +1$, i.e

$$min\ cost(\alpha; x)$$
$$s.t. \begin{cases} f(x+\alpha) = +1 \\ \alpha(x) \end{cases} \tag{2}$$

The AR method uses integer programming to solve the optimization problem in (2). This approach can search for binary, ordinal, and categorical features directly on actions; optimize nonlinear and non-convex cost functions; it allows users to customize a set of actionable actions; and can quickly find the global optimal solution, or prove that the classifier does not provide recourse.

3.2 Class imbalance

Class imbalance is the situation in binary classification in that the instance number of one class noticeably surpasses the other class. To control the class imbalance variable and ensure the credibility of the experimental comparisons, this research will select two types of datasets for experimentation. Firstly, synthetic datasets will be created based on sampling. To achieve this, a real-world churn prediction dataset will be selected at the beginning. Then data cleaning and preprocessing will be performed. To create different imbalance degrees, oversampling, and specified ratio sampling techniques will be applied to the real data. Secondly, real-world datasets from different churn prediction scenarios will be selected for experimentation.

3.3 Instance position

Instance position refers to the location of a particular instance, and in our research, we assess its proximity to the decision boundary. The KNN method will be employed to partition the instances based on their positions. For a target instance X_0, a fixed-sized neighborhood with $k = 5$ will be considered. The number of neighbors N from the opposite class as X_0 within its neighborhood will be recorded. If $N = 0$, it means all the neighbors belong to the same class as X_0 and X_0 is defined as a safe instance. Conversely, if $N = 5$, it means all the neighbors belong to the opposite class of X_0, and X_0 is defined as a boundary instance.

4 EXPERIMENTS

In this section, we will conduct comparative experiments based on the workflow outlined in Section 3. Detailed information about the data and experimental results will be provided. The experiments were conducted using Python 3.7.8, and the black-box model was built using TensorFlow 1.14.0. The counterfactual methods and evaluation metrics were obtained from the Python package carla-recourse 0.0.5. The entire experiment was run on a computer with a 1.80 GHz AMD Ryzen 7 4800U with Radeon Graphics CPU, 8.2 GB RAM, and AMD Radeon(TM) Graphics GPU.

4.1 Data set

The datasets used in the experiment consist of two parts. The first part is synthetic datasets with different imbalance levels in the same churn prediction scenario. They are obtained by oversampling a real Internet churn dataset and performing specified ratio sampling. The second part is real datasets from different churn prediction scenarios, including a bank churn data, an E-commerce churn data and a telecom data. All the original datasets used in this research are open datasets from Kaggle.

To gain a better understanding of the data, the degree of class overlap is calculated using the Fisher score as a metric. The Fisher score, also known as the F-statistic, is calculated as the ratio of between-group variance to within-group variance. It is computed for each feature individually, and the overall Fisher score is the average of the Fisher scores for all features. A higher Fisher score indicates a smaller degree of class overlap, with smaller within-class distances and larger between-class distances. The description of the data is shown in Table 1 and Table 2.

Table 1: Description of synthetic dataset.

Data set	Internet-b	Internet-c	Internet-d
Imbalance ratio	0.3	0.1	0.01
Fisher score	0.08646	0.04163	0.00618
volume	25134	35446	32546
number of features	10	10	10

Table 2: Description of the real datasets.

Data set	Bank	E-Commerce	Telecom
Imbalance ratio	0.203	0.168	0.283
Fisher score	0.01423	0.01322	0.05519
volume	10000	5630	6589
number of features	14	35	29

4.2 Evaluation

A good counterfactual explanation is such that the difference between X_0 and X_c is minimal, which means it should meet the following characteristics:

Fidelity. The primary requirement of counterfactual learning is to generate counterfactual instances that closely match the predefined prediction. However, it is not always possible to find a counterfactual with the predefined prediction in the data. For example, in a binary classification setting with high class imbalance, the model

may always classify an instance as the frequent class, making it almost impossible to change the prediction label from the majority class to the minority class by altering the feature values.

Proximity. Counterfactual instances should be as similar as possible to the instances in terms of their feature values. The distance between two instances can be measured using Manhattan distance or Euclidean distance.

Sparsity. Counterfactual instances should not only be close to the original instance but also minimize the changes in the features. To evaluate the quality of counterfactual explanations under this criterion, we can simply calculate the proportion of changed features between the counterfactual and the actual instance.

Credibility. The feature values of counterfactual instances should be reasonable. For example, generating a counterfactual explanation with a negative apartment area or setting the number of rooms to 200 is meaningless. It would be even better if the counterfactual is also plausible under the joint distribution of the data. For instance, an apartment with 10 rooms and 20 square meters should not be considered as a plausible counterfactual explanation. Ideally, if the number of square meters is increased, it should also suggest an increase in the number of rooms. This measure can be evaluated by comparing the generated counterfactual instances with their neighbors and computing the support of neighbors.

4.3 Experimental results

For each dataset, we build a two-layer artificial neural network to make predictions and set the parameters label smoothing = 1e-2 and learning rate = 1e-3. The dataset is divided into a training set and a test set. The training set is used for model training, and the classification threshold is adjusted based on the proportion of the minority class. The test set was used for testing, and ACC and AUC were used as model evaluations. The results are shown in Table 3.

Table 3: Comment on prediction models

Data set	threshold	AUC	ACC
Bank	0.8272	0.8501	0.8
E Commerce	0.919	0.944	0.7
Telecom	0.8736	0.9373667	0.7
internet-b	0.7772	0.8153	0.69
internet-c	0.89	0.8145667	0.83
internet-d	0.9829	0.7546667	0.93

The result of counterfactual explanations on evaluation matrix according to Section 4.2 is displayed in the following heat maps. The deeper the color, the higher the value of the indicator.

As shown in the Figure 2, the success rate of finding counterfactual explanations shows that the difficulty of finding counterfactual explanations varies significantly in different instance locations and data sets with imbalances. In the dataset with the highest degree of class overlap (imbalance degree 0.1861), the success rate of finding counterfactual explanations was the highest. For AR and GS methods, the success rate of safe instances (N=0) in finding counterfactual explanations is higher than that in boundary instances (N=5). The Wachter's method defaults to finding a counterfactual explanation anyway.

As shown in the Figure 3, the euclidean distance (i.e., proximity) of counterfactual explanation to the original instance is significantly different in the dataset with different instance locations and degree of imbalance. Boundary instances (N=5) find counterfactual explanations with higher proximity than safe instances (N=0). When the dataset has an imbalance of about 0.2, the counterfactual explanation found has lower proximity, and the AR method has lower proximity than the GS and Wachter methods.

As shown in the Figure 4, in different methods, there are discrepancies between counterfactual explanation and the proportion of change in the characteristics of the original instance (i.e., sparsity) in the data sets of different instance locations and degree of imbalance. In Wachter's method, boundary instances (N=5) find counterfactual explanations with higher sparsity rate than safe instances (N=0). While in AR method, the sparsity rate of counterfactual explanations of safe instances (N=0) is higher than that of boundary instances (N=5). When the data set has a smaller degree of imbalance, the counterfactual explanation has higher sparsity rate.

As shown in the Figure 5, there is a significant difference between the counterfactual explanation and the neighbour support (credibility) of the original instance in different instance locations. Boundary instances (N=5) find counterfactual explanations with better neighbor support than safe instances (N=0).

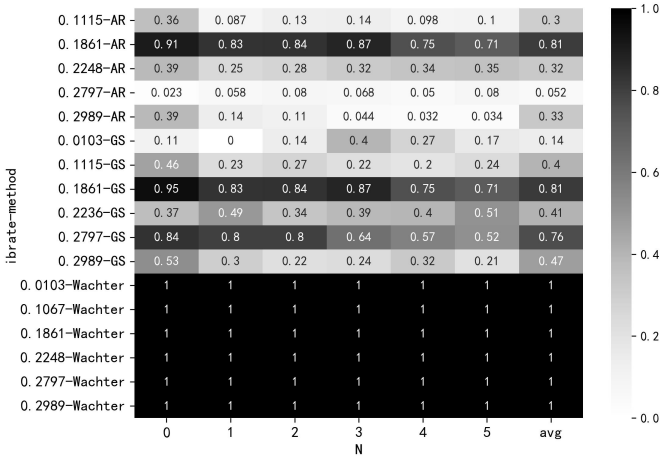

Figure 2: result of success rate

Overall, under different levels of class imbalance, there are significant differences in the success rate of counterfactual explanations, the proximity between counterfactual explanations and original instances, the proportion of feature changes (sparsity), and the support of neighboring instances to the original instance (credibility). In cases with highly class imbalance, where the number of

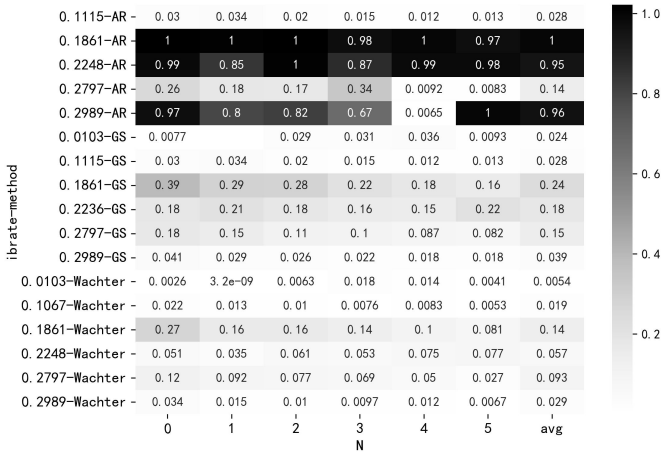

Figure 3: result of euclidean distance

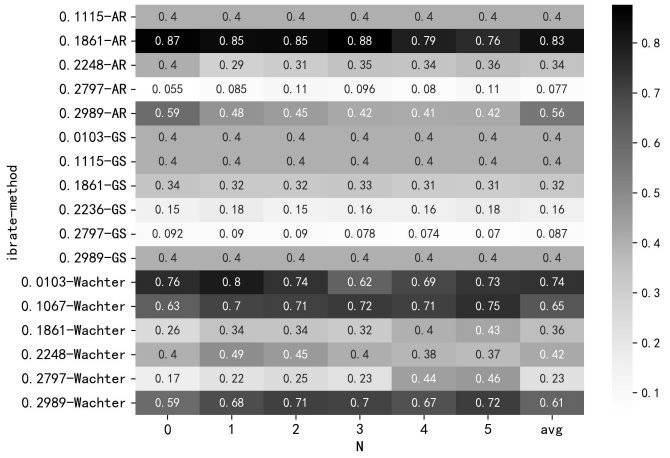

Figure 4: result of sparsity rate

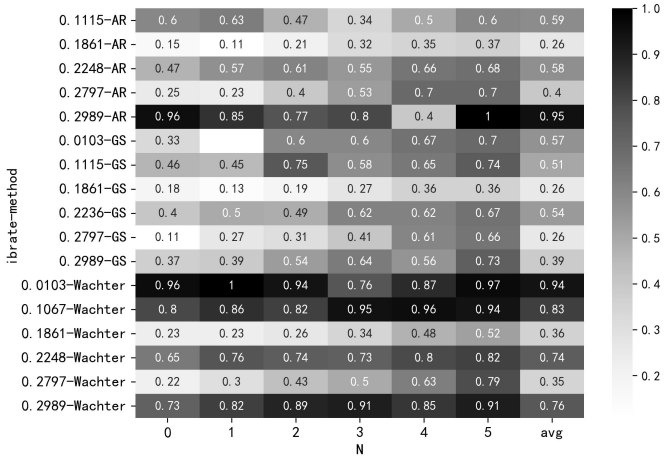

Figure 5: result of y-nearest-neighbours

instances in the minority class is extremely low, counterfactual explanations found are closer to the original data in terms of distance, involve fewer changes in features, and have better credibility. However, in cases with relatively small class imbalance, the generated counterfactual explanations differ significantly from the original instances.

The effectiveness of counterfactual learning varies depending on the position of the instances. Instances located near the decision boundary are more easily assigned counterfactual explanations compared to safe instances, and the counterfactual explanations found have better proximity, sparsity, and credibility.

The differences in counterfactual learning effectiveness due to class imbalance and instance positions vary among different counterfactual learning methods. Wachter's method ensures that each instance can find a corresponding counterfactual, but the quality of it is more sensitive to class imbalance and instance positions. It performs better in highly imbalanced scenarios and boundary instances, but relatively worse in datasets with high class overlap. The AR and GS methods are more likely to find corresponding counterfactual explanations for instances located near the decision boundary, and the GS method is more easily applied in cases with smaller class imbalance. The GS method finds counterfactuals of better quality compared to the AR method, and counterfactual explanations found by the GS method have better proximity and sparsity, especially in scenarios with high class imbalance and instances near the decision boundary.

5 CONCLUSIONS

The counterfactual explanation generated by counterfactual learning methods is an easy-to-understand and quantitative explanation of a single instance in black-box model prediction, which has application prospects in customer churn prediction. In this research, we innovatively propose that when generating counterfactual explanations, the impact of class imbalance issue and the instance location in customer data needs to be considered. Then this research experimentally proves that there are obvious differences in the success rate of finding a counterfactual explanation, the distance between counterfactual explanation and the original instance (i.e. proximity), the proportion of feature change (i.e. sparsity), and the degree of proximity support (i.e. credibility) with the original instance in different instance locations and unbalanced data sets. In scenarios with highly class imbalance and instances near the decision boundary, it is easier to find counterfactual explanations with better proximity, sparseness, and credibility. In addition, in our experiments, the AR method found a counterfactual explanation that was closer to the original example, while Wachter's method found the counterfactual explanation with the highest success rate. Future research will test more datasets, try more counterfactual learning methods, find the general law brought by the class imbalance problem, and propose corresponding solutions.

ACKNOWLEDGMENTS

This work is supported by the Fundamental Research Funds for the Central Universities (SXYPY202337).

REFERENCES

[1] Dieter Brughmans, Pieter Leyman, and David Martens. 2023. Nice: an algorithm for nearest instance counterfactual explanations. *Data Mining and Knowledge Discovery* (2023), 1–39.

[2] Carlos Carvalho, Ricardo Masini, and Marcelo C Medeiros. 2018. ArCo: An artificial counterfactual approach for high-dimensional panel time-series data. *Journal of econometrics* 207, 2 (2018), 352–380.

[3] Damien A Dablain, Colin Bellinger, Bartosz Krawczyk, David W Aha, and Nitesh V Chawla. 2022. Interpretable ML for Imbalanced Data. *arXiv preprint arXiv:2212.07743* (2022).

[4] Susanne Dandl, Christoph Molnar, Martin Binder, and Bernd Bischl. 2020. Multi-objective counterfactual explanations. In *International Conference on Parallel Problem Solving from Nature*. Springer, 448–469.

[5] Xolani Dastile, Turgay Celik, and Hans Vandierendonck. 2022. Model-agnostic counterfactual explanations in credit scoring. *IEEE Access* 10 (2022), 69543–69554.

[6] Carlos Fernández-Loría, Foster Provost, and Xintian Han. 2020. Explaining data-driven decisions made by AI systems: the counterfactual approach. *arXiv preprint arXiv:2001.07417* (2020).

[7] Riccardo Guidotti. 2022. Counterfactual explanations and how to find them: literature review and benchmarking. *Data Mining and Knowledge Discovery* (2022), 1–55.

[8] Kyosuke Hasumoto and Masayuki Goto. 2022. Predicting customer churn for platform businesses: using latent variables of variational autoencoder as consumers' purchasing behavior. *Neural Computing and Applications* 34, 21 (2022), 18525–18541.

[9] Mark T Keane, Eoin M Kenny, Eoin Delaney, and Barry Smyth. 2021. If only we had better counterfactual explanations: Five key deficits to rectify in the evaluation of counterfactual xai techniques. *arXiv preprint arXiv:2103.01035* (2021).

[10] Ramaravind Kommiya Mothilal, Divyat Mahajan, Chenhao Tan, and Amit Sharma. 2021. Towards unifying feature attribution and counterfactual explanations: Different means to the same end. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. 652–663.

[11] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. 2017. Inverse classification for comparison-based interpretability in machine learning. *arXiv preprint arXiv:1712.08443* (2017).

[12] Thibault Laugel, Marie-Jeanne Lesot, Christophe Marsala, Xavier Renard, and Marcin Detyniecki. 2018. Comparison-based inverse classification for interpretability in machine learning. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Theory and Foundations: 17th International Conference, IPMU 2018, Cádiz, Spain, June 11-15, 2018, Proceedings, Part I 17*. Springer, 100–111.

[13] David Martens and Foster Provost. 2014. Explaining data-driven document classifications. *MIS quarterly* 38, 1 (2014), 73–100.

[14] Rafael Poyiadzi, Kacper Sokol, Raul Santos-Rodriguez, Tijl De Bie, and Peter Flach. 2020. FACE: feasible and actionable counterfactual explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 344–350.

[15] Gregory Shakhnarovich, Trevor Darrell, and Piotr Indyk. 2008. Nearest-neighbor methods in learning and vision. *IEEE Trans. Neural Networks* 19, 2 (2008), 377.

[16] Berk Ustun, Alexander Spangher, and Yang Liu. 2019. Actionable recourse in linear classification. In *Proceedings of the conference on fairness, accountability, and transparency*. 10–19.

[17] Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2017. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harv. JL & Tech.* 31 (2017), 841.

[18] You Zhu, Li Zhou, Chi Xie, Gang-Jin Wang, and Truong V Nguyen. 2019. Forecasting SMEs' credit risk in supply chain finance with an enhanced hybrid ensemble machine learning approach. *International Journal of Production Economics* 211 (2019), 22–33.